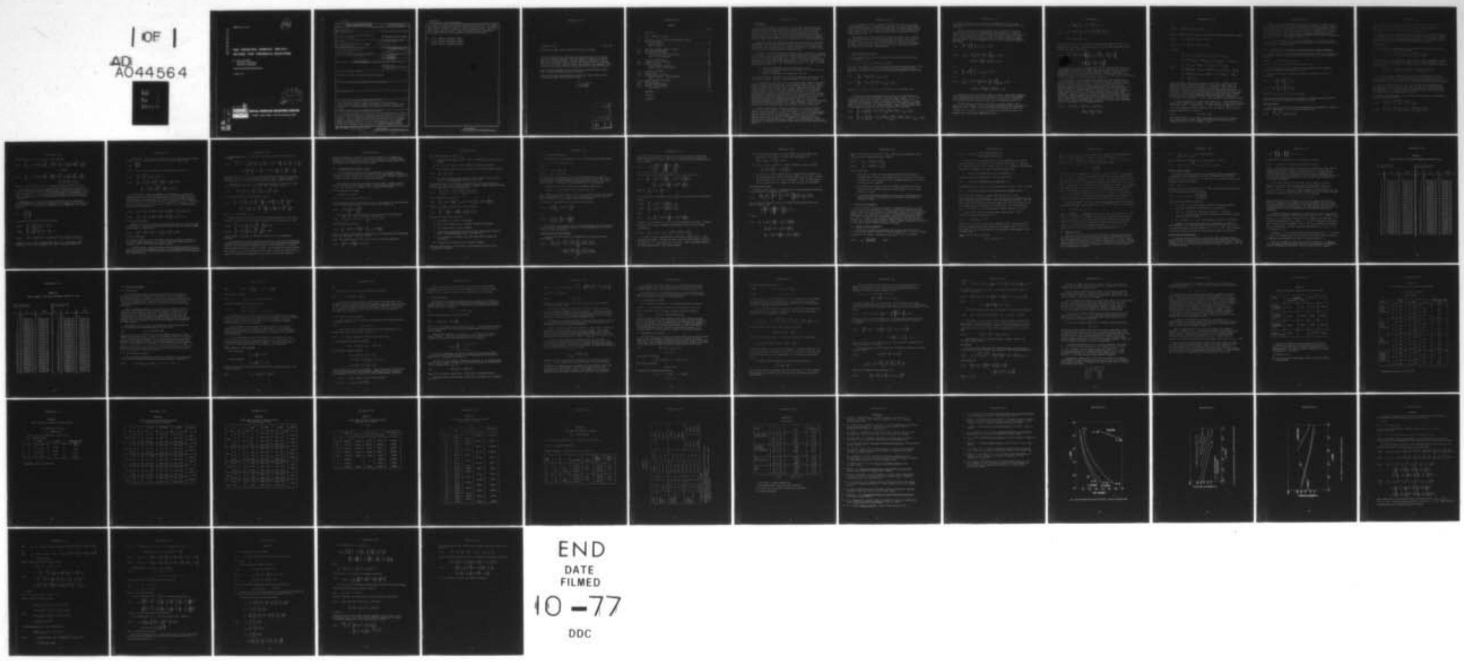
AD-A044 564 NAVAL SURFACE WEAPONS CENTER WHITE OAK LAB SILVER SP--ETC F/G 12/1
THE OPERATOR COMPACT IMPLICIT METHOD FOR PARABOLIC EQUATIONS,(U)
APR 77 M CIMENT, S H LEVENTHAL, B C WEINBERG
UNCLASSIFIED NSWC/WOL/TR-77-29 NL

1 OF 1
AD
A044564



END
DATE
FILMED
10-77
DDC

(12)

NSWC/WOL TR 77-29

THE OPERATOR COMPACT IMPLICIT METHOD FOR PARABOLIC EQUATIONS

BY MELVYN CIMENT
STEPHEN H. LEVENTHAL
BERNARD C. WEINBERG

ADVANCED WEAPONS DEPARTMENT

7 APRIL 1977

NAVAL SURFACE WEAPONS CENTER
Dahlgren, Virginia 22448 • Silver Spring, Maryland 20910

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) NSWC/WOL/TR-77-29	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) The Operator Compact Implicit Method for Parabolic Equations,		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Melvyn/Ciment, Stephen H./Leventhal, Bernard C./Weinberg		8. CONTRACT OR GRANT NUMBER(s) (16) ZR01402, WR02302
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Surface Weapons Center White Oak Laboratory White Oak, Silver Spring, Maryland 20910		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS See back (17) ZR0140201, WR0230203
11. CONTROLLING OFFICE NAME AND ADDRESS (12) 59p.		12. REPORT DATE (11) 7 April 1977
		13. NUMBER OF PAGES 55
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite difference schemes, parabolic equations, compact implicit, operator compact implicit, higher order difference schemes, boundary layer calculations, cell Reynolds number, alternating direction, factorization schemes

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper attempts to trace out the broad characteristics of a class of higher order finite difference schemes which are applicable to the solution of parabolic partial differential equations associated with viscous fluid flow problems. The basic method developed here uses the approach of the compact implicit techniques applied to the full spatial operator. The resulting spatial approximation, referred to here as the operator compact implicit method can be implemented with a variety of temporal integration schemes.

391596

In particular, a simple factorization technique is employed to resolve higher space dimension problems in terms of simple tridiagonal systems. The operator compact implicit method is compared to standard techniques and to some of the newer compact implicit methods. Stability characteristics, computational efficiency and the results of numerical experiments are discussed.

10. 61152N ; ZR00001 ; ZR0140201 ; WA0105 ;
61153N ; WR02302 ; WR02302003 ; WA8403 ;
61153N ; SR02302 ; SR02302001 ; WA8404 ;

NSWC/WOL/TR 77-29 7 April 1977

The Operator Compact Implicit Method for Parabolic Equations

The results presented in this report were obtained as part of a concerted effort to develop reliable and efficient numerical techniques for solving major fluid dynamics problems. Here the methods are designed to form the basis for a practicable computer code to solve viscous fluid flow problems. Efficient fourth order finite difference approximations are developed. Their associated stability and accuracy characteristics are analyzed and studied.

These techniques presently form the basis for a three-dimensional boundary layer computer code being developed at NSWC/WOL.

This study has been supported jointly by the Naval Surface Weapons Center Independent Research Fund, NAVAIR and NAVSEA.

C. A. Fisher

C. A. FISHER
By direction

CONTENTS

I. INTRODUCTION

The current engineering requirements for providing computational fluid dynamics codes for realistic viscous flow problems have provided the impetus for the development and implementation of higher order finite difference techniques [6], [1], [22]. It has been repeatedly demonstrated on model problems, that even the simplest types of higher order methods should provide tremendous practical advantages in terms of diminishing the required number of points (storage) and also the overall computing time for a desired resolution.

The present effort was undertaken to confront the full range of associated computational problems that would be involved in practical viscous flow field calculations. Our goal was to try to develop a cohesive set of higher order approximation tools which would help to indicate what methods ultimately might be best employed to form the basis of a major new code.

It appeared to several people almost simultaneously (sparked by a suggestion of H. O. Kreiss [14]) that from among the various techniques available a fruitful class of methods might emerge from the so-called compact implicit techniques [3], [6]. Although there appear to be a variety of forms and implementations the approaches do share some broad characteristics. The higher order is usually sought for the spatial part of the differential operator. The method developed is generally required to;

1. reduce to tridiagonal form for fourth order accuracy
2. allow for nonuniform spatial grids (usually at the expense of one order of accuracy)
3. allow for flexibility in choosing the time step.

In the various methods developed so far all these conditions have been met for simple model problems. However, further important concerns still remain.

As pointed out by [3], [4] and [2] the usual compact implicit techniques, because of their implicit complexity, are not generally applicable in a direct manner to problems with varying order derivative terms unless a vector unknown of the derivative values is considered. Indeed, adopting the factorization technique suggested in [3] for a wave equation problem to a model parabolic problem resulted in numerical instabilities (see Section III.3 below). To circumvent such problems, we advocate the use of a more general spatial approximation method, an operator compact implicit method suggested by B. Swartz [24]. Essentially, the same basic ideas are involved and instead of setting up spatial approximations for <u>individual</u> derivative terms one now poses the difference approximation in terms of the spatial <u>operator</u>. This spatial approximation has been previously derived by [17]; however, the basic derivation and implementation there proceeds along lines different from those taken here.

Another serious concern that one has relates to the stability characteristics of the overall method. If the spatial operator is associated with implicit temporal schemes, as it might be expected, a variety of unconditionally stable schemes result for the linear model. However, the cell Reynolds number stability characteristics are now somewhat more difficult to elucidate Although our analysis in section V is incomplete, all experiments to date indicate that for the operator compact implicit (OCI) approximation, there is a wider range of admissible cell Reynolds number than for the usual compact implicit methods.

In our numerical studies of nonlinear models we have chosen to use two different approaches. As a benchmark, we have taken the basic Crank-Nicolson routine solved by simple successive approximations. Our second approach adapts a Lees type method [12] which does not require temporal iterations for a non-linear problem. This latter simple scheme has proven to be very effective in numerical experiments.

What emerges from our investigation is that a promising class of methods can be developed around the operator compact implicit method. On the basis of our experiments an OCI-Lees type scheme appears to be very efficient and reliable. In the future we hope to resolve questions concerning the treatment of mixed spatial derivative terms and to more fully resolve the limitations associated with cell Reynolds number effects.

II. BASIC DIFFERENCE EQUATIONS

The classical finite difference approach for solving two-point boundary value problems of the form

$$(2.1) \qquad L(u) = a(x)\, u_{xx} + b(x)\, u_x = f, \quad x\varepsilon[0,1]$$

with $u(0)$, $u(1)$ given is to separately substitute standard approximations for the first and second derivatives in (2.1) and then solve the resulting system of equations. Accordingly, the centered second order approximation for these terms is

$$(2.2) \qquad \frac{\delta_x U_j}{2h} \equiv \frac{U_{j+1} - U_{j-1}}{2h} = (u_x)_j + O(h^2)$$

$$(2.3) \qquad \frac{\delta_x^2 U_j}{2h} \equiv \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} = (u_{xx})_j + O(h^2)$$

where $x_j = jh$, $j=0,1,\cdots,J$ and $U_j \sim u(x_j)$ and $h = 1/J$ is the mesh size.

The resulting system of equations that is derived upon substitution of (2.2), (2.3) into (2.1) is tridiagonal, and hence easily solved. For the case of Dirichlet data, there is no need to create fictitious points (i.e. to extrapolate information) in order to implement the scheme. However, if higher order accuracy is desired, the classical approach is to enlarge the basic mesh star, i.e. use more points in the discretization. Again, for the centered type of approximation fourth order accuracy is achieved by the following

$$(2.4) \qquad \left[I - \frac{1}{6}\delta_x^2\right]\frac{\delta_x U_j}{2h} = \frac{U_{j-2} - 8U_{j-1} + 8U_{j+1} - U_{j+2}}{12h} = (u_x)_j + O(h^4)$$

$$(2.5) \qquad \left[I - \frac{1}{12}\delta_x^2\right]\frac{\delta_x^2 U_j}{h^2} = \frac{-U_{j-2} + 16U_{j-1} - 30U_j + 16U_{j+1} - U_{j+2}}{12h^2} = (u_{xx})_j + O(h^4)$$

By substituting (2.4), (2.5) into (2.1) a pentadiagonal system of linear equations is obtained, and it is necessary to use fictitious points near both boundaries.

A different fourth order approximation can be obtained by following a suggestion of Kreiss [14]. The resulting representation is of an implicit nature in that there are relationships among the function and its derivative at each of three adjacent mesh points. Because the method achieves the highest order accuracy possible on the smallest star it has been called the compact implicit method. For the derivatives considered above, following our notation, one obtains

$$\text{(2.6a)} \qquad \left[I + \frac{\delta_x^2}{6}\right]^{-1} \frac{\delta_x}{2h} U_j = (u_x)_j + O(h^4)$$

or

$$\text{(2.6b)} \qquad \frac{\delta_x}{2h} U_j = \frac{U_{j+1} - U_{j-1}}{2h} = \left[I + \frac{\delta_x^2}{6}\right](u_x)_j + O(h^4)$$

$$= \frac{(u_x)_{j+1} + 4(u_x)_j + (u_x)_{j-1}}{6} + O(h^4)$$

and

$$\text{(2.7a)} \qquad \left[I + \frac{\delta_x^2}{12}\right]^{-1} \frac{\delta_x^2}{h^2} U_j = (u_{xx})_j + O(h^4)$$

or

$$\text{(2.7b)} \qquad \frac{\delta_x^2}{h^2} U_j = \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} = \left[I + \frac{\delta_x^2}{12}\right](u_{xx})_j + O(h^4)$$

$$= \frac{(u_{xx})_{j+1} + 10(u_{xx})_j + (u_{xx})_{j-1}}{12} + O(h^4)$$

Equations (2.6) and (2.7) are derivable by either a Taylor series analysis, Hermite polynomial interpolation or by thinking of (2.4) and (2.5) as Neumann series representations (up to fourth order) of (2.6) and (2.7), respectively. As a reference for these formulas in the case of an uneven grid, see [1].

By substituting (2.6) and (2.7) into (2.1) it becomes apparent that in general it is not possible to directly obtain a tractable system of equations in terms of U_j alone. Indeed, to solve the resulting system one can define new variables $F_j \sim (u_x)$ and $S_j \sim (u_{xx})_j$ and develop the following 3x3 block tri-diagonal system of equations approximating (2.1):

$$\text{(a)}\quad \frac{U_{j+1} - U_{j-1}}{2h} - \frac{F_{j+1} + 4F_j + F_{j-1}}{6} = 0$$

$$(2.8)\quad \text{(b)}\quad \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} - \frac{S_{j+1} + 10S_j + S_{j-1}}{12} = 0$$

$$\text{(c)}\quad b_jF_j + a_jS_j = f_j$$

where $b_j = b(x_j)$ and $a_j = a(x_j)$ and the above equations hold for $j=1,2\cdots,J-1$. Alternatively, omitting S_j and using only U_j, F_j a 2x2 block tridiagonal system results from using (2.8)(a) with

$$(2.9)\quad \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} + \frac{1}{12}\left(\frac{b_{j+1}}{a_{j+1}} F_{j+1} + \frac{10b_j}{a_j} F_j + \frac{b_{j-1}}{a_{j-1}} F_{j-1}\right)$$

$$= \frac{1}{12}\left(\frac{f_{j+1}}{a_{j+1}} + \frac{10f_j}{a_j} + \frac{f_{j-1}}{a_{j-1}}\right)$$

Equations (2.8) and (2.9) require more work to solve them than the second order method, but generally the higher order accuracy permits one to solve with considerably fewer points to achieve a comparable accuracy. Moreover, for Dirichlet data, no fictitious points are needed. Boundary values ($j=0,J$) are required for F_j in (2.9) and for F_j and S_j in (2.8). How these values are obtained in the time dependent case is discussed in section III.

At this point we preview some of the results that will be presented in sections III and IV where a parabolic problem with a spatial operator given by (2.1) is considered. There it will be seen that all the usual ways of solving implicit systems incorporating compact implicit schemes do not provide a generally successful method in the following sense. There does not appear to be a way of achieving a scalar tridiagonal factorization for an unconditionally stable scheme when the compact implicit schemes are used for (2.1). However, by using a different approach for the spatial operator these goals are still attainable. Namely, we abandon our attempts to represent the separate derivative terms in the spatial operator and adopt an approach which looks for a relationship on three adjacent points between L(u) and the function u. The resulting fourth order accurate relationship may be derived by a Taylor series development and can be represented in the following equations

$$(2.10a)\quad q_j^+(L(U))_{j+1} + q_j^0(L(U))_j + q_j^-(L(U))_{j-1}$$

$$= \frac{r_j^+U_{j+1} + r_j^0U_j + r_j^-U_{j-1}}{h^2}$$

or

$$\frac{Q^{-1}R}{h^2}\, u(x_j) = L(u)_j + O(h^4) \tag{2.10b}$$

where the operators Q and R are each tridiagonal displacement operators, namely

$$Q\, U_j = q_j^+\, U_{j+1} + q_j^0\, U_j + q_j^-\, U_{j-1} \tag{2.11a}$$

$$R\, U_j = r_j^+\, U_{j+1} + r_j^0\, U_j + r_j^-\, U_{j-1} \tag{2.11b}$$

and where

$$\begin{aligned}
q_j^+ &= 6a_j\, a_{j-1} + h(5a_{j-1}\, b_j - 2a_j\, b_{j-1}) - h^2\, b_j\, b_{j-1} \\
q_j^0 &= 4[15a_{j+1}\, a_{j-1} - 4h(a_{j+1}\, b_{j-1} - b_{j+1}\, a_{j-1}) - h^2\, b_{j+1}\, b_{j-1}] \\
q_j^- &= 6a_j\, a_{j+1} - h(5a_{j+1}\, b_j - 2a_j\, b_{j+1}) - h^2\, b_j\, b_{j+1} \\
r_j^+ &= \tfrac{1}{2}[q_j^+(2a_{j+1} + 3h\, b_{j+1}) + q_j^0(2a_j + hb_j) + q_j^-(2a_{j-1} - hb_{j-1})] \\
r_j^- &= \tfrac{1}{2}[q_j^+(2a_{j+1} + hb_{j+1}) + q_j^0(2a_j - hb_j) + q_j^-(2a_{j-1} - 3hb_{j-1})] \\
r_j^0 &= -\,(r_j^+ + r_j^-)
\end{aligned} \tag{2.12}$$

These relationships were first presented by Swartz [24]. Equation (2.10a) retains the scalar tridiagonal feature of a second order method while not requiring additional fictitious points at the boundary. Note, in the case where either a(x) or b(x) is identically zero, with the other coefficient identically a constant, the usual compact implicit schemes (either (2.6) or (2.7)) will result. Because of these characteristics we have adopted the terminology of referring to (2.10) as the <u>operator compact implicit</u> (OCI) method. Note a formula of structure similar to (2.10) - (2.12) is presented in Appendix A for the case of an uneven grid. In that case the method is third order accurate.

At least symbolically, we refer to the inverse of Q. The determination of when Q can be inverted is in general a difficult problem. In the case of constant coefficients ($a(x) \equiv a = \text{const}$, $b(x) \equiv b = \text{const}$) the invertibility of Q on ℓ_2 can be fully analyzed by Fourier analysis [23]. Defining $R_c = \frac{hb}{a}$ as the cell Reynolds number then Q^{-1} exists for

$$R_c \le \sqrt{12} = 3.464 \tag{2.13}$$

(The invertibility of Q on a finite dimensional space is harder to specify. For the above case, a simple sufficient condition guaranteeing diagonal dominance leads to $R_c \le (-3 + \sqrt{249})/4 = 3.195$.)

The above spatial approximation can be extracted from those schemes developed by [17], [10] in the context of approximating a time dependent parabolic operator. Their resultant system of equations is identical to an OCI approximation applied to a parabolic operator for some particular choice of a temporal scheme. However, the present approach allows one to develop a variety of combinations. Moreover, it should be observed that similar spatial approximations have been developed under various names, in particular Collatz had sometime ago advocated such approaches which he refers to as "mehrstellen" methods [5].

In order to apply any of the methods presented here to parabolic equations, it is necessary to associate a time integration method. These considerations are taken up in the following sections.

III. ALTERNATIVE TIME INTEGRATIONS WITH COMPACT IMPLICIT SPATIAL DIFFERENCES

In this section we consider time integration methods to be used in conjunction with compact implicit spatial differencing [(2.6)-(2.9)] in the model parabolic problem

$$u_t = a(x,t)\, u_{xx} + b(x,t)\, u_x \tag{3.1a}$$

$$u(0,t) = c(t);\; u(1,t) = d(t);\; u(x,0) = f(x) \tag{3.1b}$$

For a discussion of compact implicit methods applied to the comparable second order hyperbolic problem see [3], [4].

In all that follows the notation introduced in section II is used. In particular,

$$S_j^n \equiv \left[I + \frac{\delta_x^2}{12}\right]^{-1} \frac{\delta_x^2}{h^2}\, U_j^n \tag{3.2}$$

$$F_j^n \equiv \left[I + \frac{\delta_x^2}{6}\right]^{-1} \frac{\delta_x}{2h}\, U_j^n, \tag{3.3}$$

where $U_j^n \approx u(jh, n\Delta t)$ and Δt is the time step.

Where the methods presented here have appeared in a similar form in other works details are omitted and appropriate references are given.

Explicit Methods

The first class of time discretization methods to be considered are explicit methods. Discretizing (3.1) in the usual fashion yields

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = a_j^n\, S_j^n + b_j^n\, F_j^n. \tag{3.4}$$

The right hand side is evaluated by using the value of U_j^n and (3.2), (3.3). Boundary conditions for S_j^n and F_j^n must be specified. This technique was investigated by Hirsh [6], Rubin [22], [23], and Adam [1].

This method is first order accurate in time and since it is explicit in time a restrictive stability condition must be imposed (see e.g. Hirsh [23]). However, equations formed by using (3.2) and (3.3) already are of an implicit nature. Thus, essentially no extra computational work results if a second order implicit temporal method is used in order to insure unconditional stability.

Implicit Methods

Two second order unconditionally stable methods are considered. The first of these is the usual Crank-Nicolson method

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{a_j^{n+1} S_j^{n+1} + a_j^n S_j^n}{2} + \frac{b_j^{n+1} F_j^{n+1} + b_j^n F_j^n}{2} \quad (3.5)$$

The second scheme to be considered is adapted from a second order method presented by Lees [12] which used standard approximations for spatial derivatives. The Lees approach, when implemented with compact implicit spatial difference approximations, results in

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} = \frac{a_j^n (S_j^{n+1} + S_j^n + S_j^{n-1})}{3} + \frac{b_j^n (F_j^{n+1} + F_j^n + F_j^{n-1})}{3} \quad (3.6)$$

The Crank-Nicolson method would seem to be more advantageous since it is only a two-level scheme. However, observe that the coefficients in (3.6) are evaluated at the n^{th} time level, thus iteration would be unnecessary even if a or b were nonlinear. Nonlinear equations are discussed in greater detail in section VI.

Due to the implicit nature of (3.2) and (3.3) one must consider various techniques for solving (3.5) and (3.6). Here we limit our discussion to three basic approaches; predictor-corrector, block inversion and direct factorization. The implementation of the three methods is similar for each of (3.5) and (3.6), therefore details are presented only for (3.5).

1) Predictor-Corrector Method. The following predictor-corrector approximation can be used to solve (3.5).

$$\frac{\overline{U}_j^{n+1} - U_j^n}{\Delta t} = \frac{a_j^{n+1} \overline{S}_j^{n+1} + a_j^n S_j^n}{2} + b_j^n F_j^n \quad (3.7a)$$

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{a_j^{n+1} S_j^{n+1} + a_j^n S_j^n}{2} + \frac{b_j^{n+1} \overline{F}_j^{n+1} + b_j^n F_j^n}{2} \quad (3.7b)$$

Substituting (3.2) into (3.7) results in the equations

$$\text{(3.8a)}\quad \left[I - \lambda/2\, a_j^{n+1}\left[I + \frac{\delta_x^2}{12}\right]^{-1}\delta_x^2\right]\bar{U}_j^{n+1} = \left[I + \frac{\lambda}{2}a_j^n\left[I + \frac{\delta_x^2}{12}\right]^{-1}\delta_x^2\right]U_j^n + \Delta t\, b_j^n F_j^n$$

$$\text{(3.8b)}\quad \left[I - \lambda/2\, a_j^{n+1}\left[I + \frac{\delta_x^2}{12}\right]^{-1}\delta_x^2\right]U_j^{n+1} = \left[I + \frac{\lambda}{2}a_j^n\left[I + \frac{\delta_x^2}{12}\right]^{-1}\delta_x^2\right]U_j^n + \frac{\Delta t}{2}\left(b_j^{n+1}\bar{F}_j^{n+1} + b_j^n F_j^n\right),$$

where $\lambda = \Delta t/h^2$ and $\bar{F}_j^{n+1}$ in (3.8b) is formed by substituting $\bar{U}_j^{n+1}$ into (3.3).

The method may be shown to be unconditionally stable and only requires the solution of tridiagonal matrices, however, there is one serious drawback. In order to obtain a second order in time accurate method it is necessary to iterate (3.8b) several times. Due to this limitation the method is not competitive in terms of computing time. Predictor-corrector methods of this type are examined by Rubin [23].

2) Block Methods. The block tridiagonal methods fall into two categories; 3x3 block and 2x2 block inversion. Equations (3.5) combined with (3.2) and (3.3) form the system. By grouping the variables in vector format where the unknown vector is

$$\text{(3.9)}\quad \begin{pmatrix} U_j \\ F_j \\ S_j \end{pmatrix}$$

one obtains the 3x3 block tridiagonal equation

$$\text{(3.10a)}\quad \left[I + \frac{\delta_x^2}{6}\right]^{-1}\frac{\delta_x}{2h}U_j^{n+1} - F_j^{n+1} = 0$$

$$\text{(3.10b)}\quad \left[I + \frac{\delta_x^2}{12}\right]^{-1}\frac{\delta_x^2}{h^2}U_j^{n+1} - S_j^{n+1} = 0$$

$$\text{(3.10c)}\quad U_j^{n+1} - \frac{\Delta t}{2}b_j^{n+1}F_j^{n+1} - \frac{\Delta t}{2}a_j^{n+1}S_j^{n+1} = U_j^n + \frac{\Delta t}{2}b_j^n F_j^n + \frac{\Delta t}{2}a_j^n S_j^n$$

Methods of this type were investigated by Hirsh [6]. This method is also equivalent to one of the variants of the Spline 4 methods of Rubin [22].

Alternatively, substituting (3.2) into (3.5), and completing the resulting system with (3.3), and grouping the variables into the unknown vector

$$(3.11) \qquad \begin{pmatrix} U_j \\ F_j \end{pmatrix}$$

results in the 2x2 block tridiagonal system investigated by Adam [1]

$$(3.12a) \qquad \left[I + \frac{\delta_x^2}{6}\right]^{-1} \frac{\delta_x}{2h} U_j^{n+1} - F_j^{n+1} = 0$$

$$(3.12b) \qquad \left\{I - \frac{\lambda}{2} a_j^{n+1} \left[I + \frac{\delta_x^2}{12}\right]^{-1} \delta_x^2\right\} U_j^{n+1} - \frac{\Delta t}{2} b_j^{n+1} F_j^{n+1} =$$

$$\left\{I + \frac{\lambda}{2} a_j^n \left[I + \frac{\delta_x^2}{12}\right]^{-1} \delta_x^2\right\} U_j^n + \frac{\Delta t}{2} b_j^n F_j^n.$$

In using either of the block methods it is necessary to satisfy extra boundary conditions, that is, a condition for F in the 2x2 block system and conditions for F and S in the 3x3 block system. Let us illustrate how this is accomplished at the end point X=0. For the 3x3 block method three equations must be obtained in order to eliminate $F_0 = u_x|_{x=0}$ and $S_0 = u_{xx}|_{x=0}$. Combining equation (3.10a) and (3.10b) for j=2 with (3.10c) for j=1 and the independent Padé formulas (see Hirsh [6])

$$(3.13a) \qquad U_0^{n+1} - U_1^{n+1} + \frac{h}{2} (F_0^{n+1} + F_1^{n+1}) + \frac{h^2}{12} (S_0^{n+1} - S_1^{n+1}) + O(h^5) = 0$$

$$(3.13b) \qquad U_0^{n+1} - U_1^{n+1} + \frac{h}{3} (F_0^{n+1} + 2F_1^{n+1}) - \frac{h^2}{6} S_1^{n+1} + O(h^4) = 0,$$

F_0 and S_0 are eliminated.

Similarly, for the 2x2 block method two equations must be obtained in order to eliminate F_0. Combining (3.12a) and (3.12b) for j=1 and j=2 with the Hamming formula (see e.g. Ralston [18])

$$(3.14) \qquad 8U_0^{n+1} - 9U_1^{n+1} + U_3^{n+1} = \frac{3}{h} (F_0^{n+1} + 2F_1^{n+1} - F_2^{n+1}),$$

F_0 may be eliminated.

The boundary condition for the (3x3) block method was presented by Hirsh [6]. The above boundary condition can be used to retain fourth order accuracy in contrast to Adam's [1] second order boundary scheme.

In order to solve a block tridiagonal matrix the Thomas algorithm [8] is used. This algorithm is analogous to that of a tridiagonal matrix, with multiplication replaced by matrix multiplication and division replaced by multiplication of an inverse matrix.

3) Factorization Method. By directly substituting (3.2) and (3.3) into (3.5) one obtains

$$(3.15)\qquad \frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{1}{2h^2}\left[a_j^{n+1}\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\delta_x^2 U_j^{n+1} + a_j^n\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\delta_x^2 U_j^n\right]$$

$$+ \frac{1}{4h}\left[b_j^{n+1}\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\delta_x^2 U_j^{n+1} + b_j^n\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\delta_x U_j^n\right]$$

Upon examining (3.15) it is clear that there is no way to "unravel" the implicit operators. This fact has been observed by several authors (Ciment-Leventhal [3]) and indeed has caused some to abandon entirely the compact implicit methods [2].

Following the idea in [3], of completing the product, by adding the second order perturbation term, where δ_t^+ is the forward difference operator,

$$(3.16)\qquad -\frac{\Delta t^2}{4}\frac{\delta_t^+}{\Delta t}\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\frac{\delta_x^2}{h^2}\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\frac{\delta_x}{2h}U_j^n$$

to (3.15) the following factored equation results

$$(3.17)\qquad \left[I - \frac{\lambda}{2}a_j^{n+1}\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\delta_x^2\right]\left[I - \frac{\mu}{4}b_j^{n+1}\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\delta_x\right]U_j^{n+1}$$

$$= \left[I + \frac{\lambda}{2}a_j^n\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\delta_x^2\right]\left[I + \frac{\mu}{4}b_j^n\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\delta_x\right]U_j^n,$$

where $\lambda = \Delta t/h^2$ and $\mu = \Delta t/h$.

Denote the right hand side of (3.17) by G_j^n, then the solution of (3.17) can be obtained by introducing an intermediate variable Z_j^{n+1} and splitting (3.17) into

$$(3.18a)\qquad \left[I - \frac{\lambda}{2}a_j^{n+1}\left[I + \frac{1}{12}\delta_x^2\right]^{-1}\delta_x^2\right]Z_j^{n+1} = G_j^n$$

$$(3.18b)\qquad \left[I - \frac{\mu}{4}b_j^{n+1}\left[I + \frac{1}{6}\delta_x^2\right]^{-1}\delta_x\right]U_j^{n+1} = Z_j^{n+1}.$$

This technique is analogous to D'Yakonov's method [13] for two dimensional problems.

The algorithm (3.18) still requires the solution of more than one tridiagonal system, but no extra iterations are necessary as in the predictor-corrector method. However, a more fundamental difficulty persists in this formulation. An intermediate boundary condition for Z_j^{n+1} is needed. This intermediate condition implies that either u_x or u_{xx} on the boundary must be given. However specifying these in general (for example by extrapolation) could create an ill-posed problem and generate instabilities (numerical experiments conducted by the authors have revealed such instabilities). In certain problems however, e.g. boundary layer equations, from physical considerations, extra conditions may apply, then (3.18) may actually be a reliable and efficient method.

IV. THE OPERATOR COMPACT IMPLICIT METHOD

In this section we employ the OCI approximation to the spatial operator given in (2.1) with the two time discretization methods considered in section III in order to provide a method for solving the time dependent parabolic problem (3.1). The method is then extended to two dimensional problems.

The methods presented here are unconditionally stable. However, as with most other methods for this problem there is a cell Reynolds number condition (2.13). The discussion of stability will be reserved for section V.

IV.1 One-Dimensional Problems

Consider the equation

$$u_t = a(x,t)u_{xx} + b(x,t)u_x = L(u) \tag{4.1}$$

where the coefficients in (4.1) depend on time. Let n indicate the time dependence in the difference approximation to L(u) at the n^{th} time level, i.e.,

$$(LU)_j^n = \left[(Q^n)^{-1} R^n\right] \frac{U_j^n}{h^2} . \tag{4.2}$$

The first time discretization method considered here is Crank-Nicolson.

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{(Q^{n+1})^{-1} R^{n+1} U_j^{n+1} + (Q^n)^{-1} R^n U_j^n}{2h^2} , \tag{4.3}$$

which requires that one solve

$$\left[I - \lambda (Q^{n+1})^{-1} R^{n+1}\right] U_j^{n+1} = \left[I + \lambda (Q^n)^{-1} R^n\right] U_j^n , \tag{4.4}$$

where $\lambda = \Delta t/2h^2$. (Note well, for simplicity in the presentation of the equations we will be redefining λ from time to time.)

Denote the righthand side of (4.4) by G_j^n, then (4.4) can be expressed as

$$\left[Q^{n+1} - \lambda R^{n+1}\right] U_j^{n+1} = Q^{n+1} G_j^n . \tag{4.5}$$

Note the following facts about (4.5).

1) The matrix represented by $Q^{n+1} - \lambda R^{n+1}$ is tridiagonal, thus very easily solved.

2) No fictitious points, or extra boundary conditions are needed.

3) The righthand side G_j^n may be computed by the simple recurrence relation

$$G_j^n = 2U_j^n - G_j^{n-1} \tag{4.6}$$

4) The method is second order accurate in time, fourth order accurate in space, and unconditionally stable (see section V).

The second method to be considered is derived from a Lees type scheme (see section III). The Lees method combined with an operator compact implicit spatial differencing suggests the following method,

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} = \frac{(Q^n)^{-1} R^n(U_j^{n+1} + U_j^n + U_j^{n-1})}{3h^2} \tag{4.7}$$

which requires the solution of

$$\left[I - \lambda (Q^n)^{-1} R^n\right] U_j^{n+1} = \lambda (Q^n)^{-1} R^n U_j^n + \left[I + \lambda (Q^n)^{-1} R^n\right] U_j^{n-1}, \tag{4.8}$$

where now $\lambda = \frac{2\Delta t}{3h^2}$. Multiply (4.8) by Q^n to obtain

$$\left[Q^n - \lambda R^n\right] U_j^{n+1} = \lambda R^n \left[U_j^n + U_j^{n-1}\right] + Q^n U_j^{n-1} \tag{4.9}$$

Note the following facts about (4.9).

1) The matrix to be solved is tridiagonal.

2) No fictitious points or extra boundary conditions are needed.

3) The righthand side is easily computed.

4) The method is second order accurate in time, fourth order accurate in space, and unconditionally stable (see section V).

5) It is necessary to generate U_j^1 by some other method to begin the computation.

6) No iteration is necessary for a nonlinear problem.

This last point becomes very important in many applications, such as the boundary-layer equations.

IV.2 Two-Dimensional Problems

We now turn to the consideration of the two dimensional parabolic problem

(4.10) $$u_t = L_x(u) + L_y(u) \equiv L(u)$$

where

(4.11a) $$L_x(u) = au_{xx} + bu_x$$

(4.11b) $$L_y(u) = cu_{yy} + du_y$$

As pointed out in the discussion of factorization methods in section III our factorization technique can not be properly adapted with the usual compact implicit method to spatial operators with different order terms. Thus, the discussion here is restricted to the implementation of the OCI method.

For simplicity (4.10) is solved on a rectangular region given by

$$\{(x_j,y_k): \ x_j = jh_x;\ j=0,1,\cdots J,\ y_k = kh_y;\ k=0,1,\cdots K\}$$

where boundary data is prescribed for all t for j=0,J and for k=0,K, and initial data is prescribed for t=0. As in [3] it is possible to directly extend the method developed here to rectangular-like L-shaped domains.

Denote the OCI approximations to the operators in (4.11) by

(4.12a) $$\left[L_x U\right]^n_{j,k} = (Q^n_x)^{-1}\ R^n_x\ \frac{U^n_{j,k}}{h_x^2}$$

and

(4.12b) $$\left[L_y U\right]^n_{j,k} = \left[(Q^n_y)^{-1}\ R^n_y\right]\frac{U^n_{j,k}}{h_y^2}$$

The methods to be presented are of the ADI (Alternating Direction Implicit) variety and their derivations are similar to those developed in [3] for the treatment of the wave equation.

Crank-Nicolson Time Discretization

As before the first method to be examined uses a Crank-Nicolson time discretization

(4.13) $$\frac{U^{n+1}_{j,k} - U^n_{j,k}}{\Delta t} = \frac{(Q^{n+1}_x)^{-1}\ R^{n+1}_x\ U^{n+1}_{j,k} + (Q^n_x)^{-1}\ R^n_x\ U^n_{j,k}}{2h_x^2}$$
$$+ \frac{(Q^{n+1}_y)^{-1}\ R^{n+1}_y\ U^{n+1}_{j,k} + (Q^n_y)^{-1}\ R^n_y\ U^n_{j,k}}{2h_y^2}\ .$$

As in the one dimensional case where each of the derivatives was represented separately, there is no way to "unravel" the different inverse operators in (4.13). However, by adding to (4.13) the by now familiar second order perturbation cross term

$$(4.14)\qquad -\frac{\Delta t^2}{4}\frac{\delta_t^+}{\Delta t^2}\left[(Q_x^n)^{-1}\frac{R_x^n}{h_x^2}\right]\left[(Q_y^n)^{-1}\frac{R_y^n}{h_y^2}\right]U_{j,k}^n ,$$

where δ_t^+ is the forward difference operator,

the resulting equations are easily seen to assume the factored form:

$$(4.15)\qquad \left[I - \lambda_x(Q_x^{n+1})^{-1} R_x^{n+1}\right]\left[I - \lambda_y(Q_y^{n+1})^{-1} R_y^{n+1}\right]U_{j,k}^{n+1}$$

$$= \left[I + \lambda_x(Q_x^n)^{-1} R_x^n\right]\left[I + \lambda_y(Q_y^n)^{-1} R_y^n\right]U_{j,k}^n ,$$

where $\lambda_x = \dfrac{\Delta t}{2h_x^2}$ and $\lambda_y = \dfrac{\Delta t}{2h_y^2}$.

By introducing the intermediate variable $Z_{j,k}^{n+1}$ (4.15) splits into two tridiagonal systems

$$(4.16a)\qquad \left[I - \lambda_x(Q_x^{n+1})^{-1} R_x^{n+1}\right]Z_{j,k}^{n+1} = G_{j,k}^n$$

$$(4.16b)\qquad \left[I - \lambda_y(Q_y^{n+1})^{-1} R_y^{n+1}\right]U_{j,k}^{n+1} = Z_{j,k}^{n+1} ,$$

where

$$(4.17)\qquad G_{j,k}^n = \left[I + \lambda_x(Q_x^n)^{-1} R_x^n\right]\left[I + \lambda_y(Q_y^n)^{-1} R_y^n\right]U_{j,k}^n .$$

Formula (4.16) is analogous to an ADI type approximation solved with a D'Yakanov splitting. $G_{j,k}^n$ is easily computed using previous values by the following relationship

$$(4.18)\qquad G_{j,k}^n = 2(U_{j,k}^n - Z_{j,k}^n + \lambda_x(Q_x^n)^{-1} R_x^n U_{j,k}^n) + G_{j,k}^{n-1} .$$

In order to solve (4.16a) boundary conditions for $Z_{j,k}^{n+1}$ on the x = const. boundaries are needed. Likewise, in order to solve (4.16b) boundary conditions for $Z_{j,k}^{n+1}$ on the y = const. boundaries are needed. These intermediate boundary conditions are obtained in the following manner:

1) Use one sided differences to compute $Z_{j,k}^{n+1}$ at the four corner points. Here, the fact that $Z_{j,k}^{n+1}$ is a fourth order approximation to $u_{j,k}^{n+1} - \frac{\Delta t}{2}(cu_{yy} + du_y)_{j,k}^{n+1}$ is used.

2) On the x = const. boundaries (4.16b) is employed to solve for $Z_{j,k}^{n+1}$

$$Q_y^{n+1}\ Z_{j,k}^{n+1} = \left[Q_y^{n+1} - \lambda_y\ R_y^{n+1}\right]U_{j,k}^{n+1}$$

3) Now that the x = const. boundary data for $Z_{j,k}^{n+1}$ have been obtained, one can proceed with the x sweeps of the ADI scheme using (4.16a). Included in these sweeps are the y = const. boundaries. Thus, the $Z_{j,k}^{n+1}$ boundary values necessary for the y sweeps in (4.16b) are now fully available.

Lees Time Discretization

Finally, a method which is a generalization of the one dimensional OCI-Lees scheme is examined. Approximate (4.10) by

$$(4.19)\qquad \frac{U_{j,k}^{n+1} - U_{j,k}^{n-1}}{2\Delta t} = \left[(Q_x^n)^{-1}\ \frac{R_x^n}{h_x^2} + (Q_y^n)^{-1}\ \frac{R_y^n}{h_y^2}\right]\frac{(U_{j,k}^{n+1} + U_{j,k}^{n} + U_{j,k}^{n-1})}{3}$$

Again, in order to obtain a factored tri-diagonal method one adds the second order perturbation term

$$-\frac{4\Delta t^2}{9}\left[(Q_x^n)^{-1}\ \frac{R_x^n}{h_x^2}\right]\left[(Q_y^n)^{-1}\ \frac{R_y^n}{h_y^2}\right]\frac{\delta_t}{\Delta t}\ U_{j,k}^n$$

to obtain

$$(4.20)\qquad \left[I - \lambda_x(Q_x^n)^{-1}\ R_x^n\right]\left[I - \lambda_y(Q_y^n)^{-1}\ R_y^n\right]U_{j,k}^{n+1}$$

$$= \left[\lambda_x(Q_x^n)^{-1}\ R_x^n + \lambda_y(Q_y^n)^{-1}\ R_y^n\right]U_{j,k}^n$$

$$+ \left[I + \lambda_x(Q_x^n)^{-1}\ R_x^n\right]\left[I + \lambda_y(Q_y^n)^{-1}\ R_y^n\right]U_{j,k}^{n-1}$$

Denote the righthand side of (4.20) by $G^n_{j,k}$, introduce an intermediate value $Z^{n+1}_{j,k}$ and apply a D'Yakanov splitting to obtain

$$(4.21a) \qquad [Q^n_x - \lambda_x R^n_x]Z^{n+1}_{j,k} = Q^n_x G^n_{j,k}$$

$$(4.21b) \qquad [Q^n_y - \lambda_y R^n_y]U^{n+1}_{j,k} = Q^n_y Z^{n+1}_{j,k}$$

Note the following:

1) There does not appear to be any simple algorithm for computing the righthand side. However, upon multiplying $G^n_{j,k}$ by Q^n_x (as in (4.21a)) it is clear that only a backsolve of the tridiagonal matrix Q^n_y for different righthand sides is required.

2) The intermediate boundary condition for $Z^{n+1}_{j,k}$ is obtained in the same manner as in the Crank-Nicolson case once the $Z^{n+1}_{j,k}$ at the four corner points are computed.

3) As in the one-dimensional problem an extra plane of information must be generated to begin the computation and no iteration is necessary for nonlinear problems.

V. STABILITY CONSIDERATIONS

In this section we discuss two stability characteristics which enter into the evaluation of the usefulness of difference schemes for parabolic equations. At the threshold one must consider the Lax-Richtmyer stability of the evolutionary operator [19]. More recently, it has come to be appreciated that the stability characteristics associated with the spatial operator should be examined [20], [7], [11]. The ability of a spatial difference scheme to resolve the spatial variation in a region of sharp gradients (boundary layer) often gives rise to a so called cell Reynolds number condition. Here we examine these stability questions for the compact implicit schemes previously discussed.

V.1. Temporal Stability Analysis

For the case of constant coefficients one can analyze the L_2 stability of the difference scheme of interest by Fourier analysis [19]. Here the discussion is limited to OCI schemes. Substituting $U^n_j = \rho^n_{CN} e^{ij\theta}$ into (4.3) yields

$$(5.1a) \qquad \rho_{CN} = \left(\frac{2+\lambda\ell(\theta)}{2-\lambda\ell(\theta)}\right) \qquad \text{where}$$

$$(5.1b) \qquad \ell(\theta) = 3a \frac{24(\cos\theta-1)+iR_c(12-R_c^2)\sin\theta}{30-2R_c^2+(6-R_c^2)\cos\theta+i3R_c\sin\theta}$$

The term $\ell(\theta)$ is associated with the Fourier transform of the spatial operator alone [24]. For stability is is required that $|\rho_{CN}^2| \leq 1$. Imposing this condition directly on (5.1) yields, $\mathrm{Re}\,\ell(\theta) \leq 0$ as a necessary and sufficient condition for stability. This latter condition requires that

$$24(\cos\theta-1)[30-2R_c^2+(6-R_c^2)\cos\theta] + (12-R_c^2)3R_c^2\sin^2\theta \leq 0 \ .$$

Collecting terms and factoring out a $(\cos\theta-1)$ term yields

$$(5.2) \qquad (\cos\theta-1)[720-84R_c^2+3R_c^4+\cos\theta(144-60R_c^2+3R_c^4)] \leq 0$$

Regrouping, and noting from (2.13) that the region of interest is $R_c^2 \leq 12$, yields

$$(5.3) \qquad (\cos\theta-1)[12(12-R_c^2)+12(12-R_c^2\cos\theta)+288+(144-72R_c^2+3R_c^4)(\cos\theta+1)] \leq 0 \ .$$

To see that this inequality is always satisfied for $R_c^2 \leq 12$, note that the term in the left parentheses is ≤ 0 and the term in the bracket is the sum of four terms, the first three of which are *clearly non-negative*. The last term in the bracket takes on a negative minimum at $R_c^2 = 12$ and even when $\cos\theta=1$ this minimum is just the negative of the third term. This establishes that $|\rho_{CN}| \leq 1$ for $R_c^2 \leq 12$, and the unconditional temporal stability of OCI-CN.

To see that OCI-Lees is similarly stable, substitute $U_j^n = \rho_L^n e^{ij\theta}$ into (4.7) to obtain a quadratic for ρ_L,

$$(5.4) \qquad \rho_L^2 + \frac{1}{2}(K+1)\rho_L + K = 0$$

where $K = \rho_{CN}$ as in (5.1) above (with λ replaced by $\frac{2}{3}\lambda$). Since the OCI-CN method is unconditionally stable, clearly in the range of $R_c^2 \leq 12$, $|K| \leq 1$. The stability of the OCI-Lees method is now contained in the statement of the following lemma.

<u>Lemma</u>. For the roots ρ_L of (5.4)

$$|\rho_L| \leq 1 \text{ iff } |K| \leq 1.$$

Proof. First we prove the lemma for the case of equality in both inequalities. Say $K = e^{i\phi}$ then solve for ρ_L directly as $\rho_L = \bar{\rho}\, e^{i\phi/2}$ where $\bar{\rho} = e^{\pm i\psi}$, $\cos\frac{\phi}{2} = -2\cos\psi$. Clearly such ψ exists and thus $|\rho_L| = |\bar{\rho}| = 1$. On the other side, if $|\rho_L| = 1$, say $\rho_L = e^{i\phi/2}$, then solving for K yields

$$K = -\frac{1+2e^{i\phi/2}}{1+2e^{-i\phi/2}}$$

Thus $|K| = 1$. This completes the proof that $|\rho_L| = 1$ iff $|K| = 1$. To show that $|\rho_L| < 1$ iff $|K| < 1$ examine the variation of the roots ρ_L with respect to the unit circle as K varies from 0 to $+\infty$. At $K=0, \rho_L = 0, -\frac{1}{2}$, both roots are inside the unit circle. Now by a connectivity argument, and the fact that the ρ_L roots depend continuously on the coefficient K [15], varying K such that $|K| < 1$ then the corresponding roots ρ_L must remain strictly inside the unit circle. Indeed, if some root "touched" the unit circle, i.e. $|\rho_L| = 1$, then by our proof above $|K| = 1$. This argument demonstrates that for $|K| < 1$, $|\rho_L| < 1$. Conversely at $K = +\infty$ both ρ_L roots are outside the unit circle thus, again by a connectivity argument the ρ_L must remain outside the unit circle for all K such that $|K| > 1$.

Finally the stability of the two dimensional Lees-OCI method is established using the above Lemma. Substituting $U^n_{j,k} = \rho^n e^{i(j\theta + k\phi)}$ into (4.20) one obtains

$$\rho^2 + \frac{1}{2}(1 - \alpha\beta)\rho - \alpha\beta = 0 \tag{5.5}$$

where $\alpha = \dfrac{1+\lambda x\ell(\theta)}{1-\lambda x\ell(\theta)}$, $\beta = \dfrac{1+\lambda y\ell(\theta)}{1-\lambda y\ell(\theta)}$ and where $\ell(\theta)$ and $\ell(\phi)$ are defined by (5.1b) for x and y, respectively. Noting that α,β are each separately in the form of a ρ_{CN} as found above, one concludes from the above lemma, that in the range $|R^x_c| \le \sqrt{12}$, $|R^y_c| \le \sqrt{12}$ (i.e. where the cell Reynolds number invertibility condition is satisfied for each spatial operator) $|\alpha| \le 1$, $|\beta| \le 1$. Now identifying $K = -\alpha\beta$ in (5.5) clearly our above Lemma implies $|\rho| \le 1$.

V.2 Spatial Stability

Experience with computations involving diffusion convection equations has long shown that nonphysical oscillations will appear in the computed solution when the spatial mesh size is not sufficiently small [20], [7], [11]. Here we use the standard linear analysis to attempt to predict some of the cell Reynolds number limitations associated with the methods discussed in this paper. Throughout this subsection, for discussion purposes, we will consider the following model "boundary layer" problem

$$(5.6) \qquad au_{xx} - bu_x = 0\ , \quad a,b \text{ positive constants}$$
$$u(0) = 0\ , \quad u(1) = 1$$

where in general b/a is large. Note, the solution of (5.6) is

$$(5.7) \qquad u(x_j) = c_1 + c_2\, e^{\frac{b}{a}x} = c_1 + c_2\, e^{R_c j}\ , \quad x_j = j\Delta x.$$

Operator Compact Implicit

The spatial stability analysis for this method is quite straightforward and provides a practical guide for the range of usefulness of the scheme. Assuming $Q^{-1}RU_j = 0$ is applied to (5.6) then one is to consider the three point homogeneous difference equation

$$(5.8) \qquad RU_j = 0$$

Substituting a solution of the form $U_j = \mu^j$ into (5.8) (using (2.11b)) leads to the following general difference solution

$$(5.9) \qquad U_j = c_1 + c_2\mu^j;\ \mu = \frac{24+R_c(12-R_c^2)}{24-R_c(12-R_c^2)}$$

Three cases are possible for general R_c:

1. $R_c < \sqrt{12}$, $\mu > 1$. The difference solution is monotone increasing, concave up and properly approximates the true solution

2. $\sqrt{12} < R_c < 4.207607$ (R_c value where numerator of μ vanishes), $0 < \mu < 1$. The difference solution is monotone increasing but concave down and completely wrong.

3. $R_c > 4.207607$, $-1 < \mu < 0$. The difference solution is oscillatory.

In summary, the spatial modal analysis, of essentially the operator R indicates that the cell Reynolds number R_c should be restricted to the exact same condition used for the invertibility of Q, i.e. $R_c \leq \sqrt{12}$. This represents no additional limitation on how one would prudently employ the OCI method.

Compact Implicit-Block Methods

To check the spatial stability of any of the block tridiagonal compact implicit methods it is sufficient to consider any one of them since each method (either the 2x2, or the 3x3) has the same set of characteristic roots. Thus, the fundamental modes of the system can be obtained by taking a solution to (2.8 a,c) of (5.6) of the form

$$(5.10)\qquad \begin{pmatrix} U_j \\ F_j \end{pmatrix} = \mu^j \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \quad j=0,1,\cdots,J.$$

A nontrivial solution results if the determinantal equation

$$(5.11)\qquad (\mu-1)[(4-R_c)\mu^3+(12-11R_c)\mu^2-(12+11R_c)\mu-(4+R_c)]=0$$

holds. A study of (5.11) will at least provide an indication of what types of nonphysical results are possible. However, there are four roots (and corresponding arbitrary constants) to contend with now. A proper analysis involves consideration of the particular schemes used to approximate the required derivatives at the boundaries. Here we present a qualitative analysis of the possible numerical solutions of (5.6) along with some illustrative computational experiments.

For our model example (5.6) one would like to obtain a U_j which is monotone, or at least, does not have large oscillatory modes which are dominant. Generally, this is accomplished by restricting R_c so that if Re $\mu < 0$ then $|\mu| \leq 1$. However, a simple inspection of the bracketed cubic in (5.11) at $\mu = -1,0,1$ reveals that such a condition can not be found, since there are always three real roots of (5.11), μ_+, μ_-, μ_0 such that

$$\mu_+ > 1,\ \mu_- < -1,\ -1 < \mu_0 < 0.$$

Thus the block tridiagonal schemes for (5.6) do not satisfy what has been generally considered as a reasonable stability requirement. Yet the schemes are useful in practice, see section VI. The reason why the oscillatory modes do not even appear in some calculations, let alone dominate them, is tied to a consideration of the way the coefficients are determined by the boundary conditions.

A series of numerical experiments was made for (5.6) and qualitatively we can conclude the following. In the range of R_c values $(0 < R_c < 4/\sqrt{15} = 1.0328)$ where $\mu_+ \leq |\mu_-|$ no dominant oscillations occur. While in the range $\frac{4}{\sqrt{15}} \leq R_c \leq 2.14383$ corresponding to the μ_+ range $|\mu_-| \leq \mu_+ \leq e^{R_c}$ the negative oscillations tend to affect more of the region. For $\mu_+ > e^{R_c}$ the oscillations are apparent in most of the region. Typical results are presented for $R_c = 1.0, 1.5, 2.0, 2.4$ in Tables 5.1, 5.2.

The case $R_c = 2.4$ is particularly interesting because here $\mu_- = -2$ and the μ_- term is the dominant term in the solution in the interior part of region as is apparent by observing that the ratio of successive terms is -2.

Since the circumstances where these spatial oscillations will dominate (they are always present in general) is not easily anticipated, one should be aware of this potential problem for the block compact implicit methods.

Table 5.1

Compact Implicit (2x2) Block Tridiagonal Solution of (5.6)

R_c = 1.0, b/a = 30.			R_c = 1.5, b/a = 45.		
j	U_j	$u(x_j)$	j	U_j	$u(x_j)$
1	0.	0.	1	0.	0.
2	.16513E-12	.16079E-12	2	.26899E-13	.99664E-19
3	.61378E-12	.59786E-12	3	-.50788E-13	.54633E-18
4	.18330E-11	.17860E-11	4	.14292E-12	.25481E-17
5	.51414E-11	.50155E-11	5	-.33484E-12	.11520E-16
6	.14134E-10	.13794E-10	6	.84259E-12	.51727E-16
7	.38533E-10	.37658E-10	7	-.20584E-11	.23192E-15
8	.10486E-09	.10253E-09	8	.50913E-11	.10395E-14
9	.28480E-09	.27885E-09	9	-.12520E-10	.46589E-14
10	.77395E-09	.75816E-09	10	.30904E-10	.20880E-13
11	.21010E-08	.20611E-08	11	-.75975E-10	.93576E-13
12	.57087E-08	.56027E-08	12	.18793E-09	.41938E-12
13	.15495E-07	.15230E-07	13	-.45992E-09	.18795E-11
14	.42104E-07	.41399E-07	14	.11474E-08	.84235E-11
15	.11428E-06	.11254E-06	15	-.27645E-08	.37751E-10
16	.31053E-06	.30590E-06	16	.70919E-08	.16919E-09
17	.84279E-06	.83153E-06	17	-.16225E-07	.75826E-09
18	.22903E-05	.22603E-05	18	.45561E-07	.33983E-08
19	.62155E-05	.61442E-05	19	-.87359E-07	.15230E-07
20	.16892E-04	.16702E-04	20	.32645E-06	.68256E-07
21	.45839E-04	.45400E-04	21	-.30805E-06	.30590E-06
22	.12458E-03	.12341E-03	22	.29748E-05	.13710E-05
23	.33806E-03	.33546E-03	23	.25611E-05	.61442E-05
24	.91885E-03	.91188E-03	24	.37845E-04	.27536E-04
25	.24931E-02	.24788E-02	25	.10386E-03	.12341E-03
26	.67769E-02	.67379E-02	26	.62403E-03	.55308E-03
27	.18386E-01	.18316E-01	27	.23905E-02	.24788E-02
28	.49982E-01	.49787E-01	28	.11645E-01	.11109E-01
29	.13560E+00	.13534E+00	29	.49587E-01	.49787E-01
30	.36864E+00	.36788E+00	30	.22727E+00	.22313E+00
31	.10000E+01	.10000E+01	31	.10000E+01	.10000E+01

Table 5.2

Compact Implicit (2x2) Block Tridiagonal Solution of (5.6)

R_c = 2.0, b/a = 60.			R_c = 2.4, b/a = 72.		
j	U_j	$u(x_j)$	j	U_j	$u(x_j)$
1	0.	0.	1	0.	0.
2	.40527E-11	.55946E-25	2	.10396E-09	.53927E-30
3	-.60778E-11	.46933E-24	3	-.13186E-09	.64837E-29
4	.16219E-10	.35239E-23	4	.34476E-09	.72010E-28
5	-.32360E-10	.26094E-22	5	-.60936E-09	.79432E-27
6	.73391E-10	.19287E-21	6	.12990E-08	.87565E-26
7	-.15680E-09	.14252E-20	7	-.25178E-08	.96525E-25
8	.34427E-09	.10531E-19	8	.51159E-08	.10640E-23
9	-.74643E-09	.77811E-19	9	-.10152E-07	.11729E-22
10	.16277E-08	.57495E-18	10	.20383E-07	.12929E-21
11	-.35401E-08	.42484E-17	11	-.40686E-07	.14252E-20
12	.77089E-08	.31391E-16	12	.81453E-07	.15710E-19
13	-.16777E-07	.23195E-15	13	-.16283E-06	.17317E-18
14	.36522E-07	.17139E-14	14	.32573E-06	.19089E-17
15	-.79496E-07	.12664E-13	15	-.65138E-06	.21042E-16
16	.17304E-06	.93576E-13	16	.13028E-05	.23195E-15
17	-.37667E-06	.69144E-12	17	-.26056E-05	.25569E-14
18	.81991E-06	.51091E-11	18	.52113E-05	.28185E-13
19	-.17847E-05	.37751E-10	19	-.10423E-04	.31068E-12
20	.38851E-05	.27895E-09	20	.20845E-04	.34247E-11
21	-.84541E-05	.20612E-08	21	-.41690E-04	.37751E-10
22	.18423E-04	.15230E-07	22	.83381E-04	.41614E-09
23	-.39949E-04	.11254E-06	23	-.16676E-03	.45872E-08
24	.88083E-04	.83153E-06	24	.33357E-03	.50565E-07
25	-.18344E-03	.61442E-05	25	-.66651E-03	.55739E-06
26	.46035E-03	.45400E-04	26	.13401E-02	.61442E-05
27	-.55256E-03	.33546E-03	27	-.26014E-02	.67729E-04
28	.45128E-02	.24788E-02	28	.60827E-02	.74659E-03
29	.14551E-01	.18316E-01	29	-.23290E-02	.82297E-02
30	.14782E+00	.13534E+00	30	.11462E+00	.90718E-01
31	.10000E+01	.10000E+01	31	.10000E+01	.10000E+01

VI. NUMERICAL EXPERIMENTS

VI.I Introduction

In this section results of numerical experiments conducted with the various schemes that were discussed in sections II - IV are presented. These calculations were performed in order to determine the viability of the OCI method for solving viscous flow problems and to understand its characteristics and limitations and to compare its performance with the classical second order techniques now in general use as well as to other fourth order approaches.

One of our major concerns is the efficiency of the various schemes, i.e. computation time required to obtain a given accuracy. Obviously this is machine as well as programmer dependent. In order not to bias any of the techniques care was taken to program the algorithms in an efficient and consistent manner. The computing times that are given include time for: matrix setups, inversions, boundary condition evaluations and (for nonlinear problems) iteration procedures. All results were computed on the NSWC/WOL CDC 6500 computer.

The operation count estimates (multiplications and divisions) for the block tridiagonal inversion algorithm is given in [8] as

$$\text{ops} = (3n-2)(m^3 + m^2) \tag{6.1}$$

where m is the order of the block and n is the number of equations. This estimate assumes full blocks. However, if the specific values of the elements of the blocks are taken into account, e.g. zeroes and ones, the actual operation count can be greatly reduced. A comparison of operation counts for the various inversion procedures (assuming full blocks) and the modified algorithms are presented in Table 6.1. Also included there are the counts for matrix setup operations. Note that for the block methods the inversion of the matrix is the dominant factor in the running time, while for the OCI technique the matrix setup accounts for most of the time.

VI.2 Linear Parabolic Equation

The first numerical experiment involved the solution of a one dimensional linear parabolic partial differential equation with variable coefficients

$$u_t = a(x,t)u_{xx} + b(x,t)u_x \ ; \quad t \geq 0 \ ; \quad 0 \leq x \leq 1, \tag{6.2a}$$

where $$a(x,t) = \frac{1}{2}\frac{(x+1)}{(t+2)^2} \ ; \quad b(x,t) = \frac{1}{2}\frac{(x+1)}{(t+2)}$$

with the exact solution

$$u(x,t) = u_e(x,t) = \exp[(x+1)(t+2)] \tag{6.2b}$$

Initial and boundary conditions are given by

$$\begin{aligned} &u(x,0) = u_e(x,0) \\ &u(0,t) = u_e(0,t) \ ; \quad u(1,t) = u_e(1,t) \end{aligned} \tag{6.2c}$$

This example was constructed in order to test the stability and convergence properties of the methods under consideration for a variable coefficient problem. Results are shown in Table 6.2 and Figure 1. All the methods tested were stable and show the predicted convergence rates. Crank Nicolson temporal integration was used for all the schemes.

Of basic interest is the savings that can be obtained in storage and computational time. As noted in Table 6.2 and Figure 1 the OCI technique is the most efficient scheme of the methods tested. This is not wholly unexpected since the block methods require additional work to compute the first and/or second derivatives.

It is also important to note the differences in the computed L_2 errors of the fourth order methods. These result from several factors among which are the local truncation error and boundary conditions. The spatial truncation errors, which are dominant for the case considered, are given below.

Compact Implicit - (Block Methods)

First derivative

$$E_f = \frac{-h^4}{180} u^{v} + O(h^6)$$

Second derivative

$$E_s = \frac{-h^4}{240} u^{vi} + O(h^6)$$

Thus for equation (2.1) with a and b constant the local spatial truncation error at point j would be

$$E = -h^4 \left(\frac{a}{240} u_j^{vi} + \frac{b}{180} u_j^{v}\right) \tag{6.3}$$

OCI

Specializing Eq. (A.16) for constant coefficients yields

$$E = - h^4 \left(\frac{a}{200} u_j^{vi} + \frac{3b}{200} u_j^{v}\right) \tag{6.4}$$

In achieving a scalar tridiagonal system, the OCI technique leads to an unsymmetric difference formula and thus has a larger local truncation error than the block methods that were derived from symmetric formulations. Were it not for the different boundary conditions, Pade relations (3.13) for the 3x3 blocks and a Hamming type formula for the 2x2 blocks (3.14) both block methods would give identical errors.

VI.2.1 General Boundary Conditions

The OCI method can also be applied to problems with more general boundary conditions of the form

$$A\, u_x + Bu = g \tag{6.5}$$

A linear fourth order accurate expression is sought relating u_x at the boundary with u and L(u) at points j=0,1,2, i.e.

$$F_0 \equiv (u_x)_0 = H_0U_0 + H_1U_1 + H_2U_2 + G_0f_0 + G_1f_1 + G_2f_2 \tag{6.6}$$

Employing the differential equation

$$L(u)_j = aS_j + bF_j = f_j \qquad (j=0,1,2) ,$$

and the compact implicit formulas

$$F_0 + 4F_1 + F_2 = \frac{3}{h} (U_2 - U_0)$$

$$S_0 + 10S_1 + S_2 = \frac{12}{h^2} (U_0 - 2U_1 + U_2)$$

$$S_0 + 4S_1 + S_2 = \frac{3}{h} (F_2 - F_0) ,$$

the coefficients in (6.6) can be evaluated. These coefficients, the truncation error, and the extension to time dependent problems are given in Appendix B. As an example, Eq. (6.2a) was solved with the following boundary conditions

$$@\ x = 0 , \quad u + u_x = u_e(0) + u_{e_x}(0) = (t+3) \exp [t+2]$$

$$@\ x = 1 , \quad u = u_e(1) = \exp [2(t+2)]$$

Table 6.3 shows the L_2 errors and L_2 rates for different mesh widths. Comparisons with the results in table 6.2 indicate that for general boundary conditions the L_2 error is larger and the computation time is increased.

VI.3 Burgers Equation

The results just presented although extremely promising are obtained for a linear equation. In order to investigate the various methods for a nonlinear problem that might be indicative of viscous flows the one dimensional Burgers equation was chosen. Consider

$$u_t = -(u-\alpha)u_x + \nu u_{xx} \tag{6.7}$$

With the exact steady state solution given by

$$u_e(x) = \alpha\left\{1 - \tanh\left(\frac{\alpha x}{2\nu}\right)\right\} \tag{6.8}$$

Near $x = 0$, $u(x)$ exhibits large gradients, and as $\nu \to 0$ a steep shock wave forms. The ability to resolve this flow field would demonstrate the viability of the various methods.

Solutions were obtained in the domain $-5 \leq x \leq 5$ with $\alpha = 1/2$ and for various values of ν, and with the exact values of $u(x)$ specified at the boundaries. The initial conditions employed for all cases are

$$u(x,0) = \begin{cases} 1 & -5 < x < 0 \\ .5 & x = 0 \\ 0 & 0 < x < 5 \end{cases}$$

Results of computations with the OCI (Crank Nicolson and Lees) methods and the second order Crank Nicolson finite difference scheme are presented in Tables 6.4, 6.5, 6.6 and 6.7 and Figure 2.

Since Eq. (6.5) is nonlinear, iteration is necessary for the Crank-Nicolson temporal discretization. We adapt the OCI method with successive approximation for the nonlinear term, uu_x, i.e.

$$U_j^{n+1}(U_x)_j^{n+1} = U_j^{*}(U_x)_j^{n+1} \tag{6.9}$$

where U_j^{*} is the latest iterant value. This procedure converges linearly.

The second order finite difference scheme uses a different type of linearization, i.e.

(6.10a) $$(U_j - \alpha)^{n+\frac{1}{2}} (U_x)_j^{n+\frac{1}{2}} = \frac{(U_j^{n+\frac{1}{2}} - \alpha)}{2} \left\{ \frac{U_{j+1}^{n+1} - U_{j-1}^{n+1}}{2\Delta x} + \frac{U_{j+1}^{n} - U_{j-1}^{n}}{2\Delta x} \right\}$$

where $U_j^{n+\frac{1}{2}}$ is replaced by

(6.10b) $$(U_j^* + U_j^n)/2 \ ,$$

U_j^* being the lastest iterant. This form of iteration has super-linear convergence properties [16].

Both methods assume an initial guess for $U_j^{n+1} = U_j^*$ which is used to solve the resultant tridiagonal system of equations. Iteration is employed until the difference between successive iterants is less than some preset tolerance. The steady state is assumed when differences in solution values at two time steps is less than some predetermined value.

In contrast to the above procedure, the OCI-Lees discretization does not require iteration and generally approached the steady state in about the same number of time steps as the OCI-CN method.

Figure 2 presents a graph of the computed L_2 error versus the number of intervals, for the fourth order and second order schemes. The storage savings possible with the OCI method are readily evident from the figure. Tables 6.6 and 6.7 compare solution values obtained from the 4th order and second order methods with the exact value, for two cases, $\nu = .5$ and $\nu = .031$.

Although the cell Reynolds number analysis for the OCI method given in section V was derived for a linear spatial operator, this theory can be useful to predict the behavior for nonlinear time dependent problems. For the Burgers equation it was found that physical solutions were obtained for a steady state only when $|R_c|_{max} < 2.55$, where

$$|R_c|_{max} = \frac{(u-\alpha)\Delta x}{\nu} = \frac{\Delta x}{2\nu}$$

However, a careful inspection of the numerical results indicates that for $|R_c|_{max} > 2.55$, in computing the transient solution, values are obtained which yield cell Reynolds numbers exceeding $\sqrt{12}$, and physical steady state solutions can not be obtained. These results suggest that when the homogeneous case maintains, one monitor the evolution of the local cell Reynolds number and consider modifying the spatial mesh when necessary.

In contrast to the above behavior, for the boundary layer equations oscillations when they occurred were confined to some local region, but physical solutions were still obtainable elsewhere (see section VI.5).

The results of the computations presented above indicate that the OCI method can be adapted to handle nonlinearities with very little additional effort and can resolve regions with sharp gradients.

VI.4 Two Dimensional Problems

The OCI method was tested for a two dimensional parabolic equation

$$u_t = a(x,y,t)u_{xx} + b(x,y,t)u_x + c(x,y,t)u_{yy} + d(x,y,t)u_y \tag{6.11a}$$

whose coefficients were constructed in order to obtain the solution

$$u(x,y,t) = \exp\{(x+1)(y+1)(t+1)\}. \tag{6.11b}$$

Neither efficiency studies nor comparisons with other methods were made. The aim here was mainly to check the order of accuracy and the viability of the ADI formulation. Table 6.9 demonstrates that the splitting technique given in section IV yields fourth order accuracy. Ciment and Leventhal [3] have demonstrated that for hyperbolic equations this type of ADI scheme retains fourth order accuracy on other than rectangular domains, e.g. L shaped domains. Similar results are expected for parabolic equations.

VI.5 Boundary Layer Equations

The main thrust of this work is to develop methods that could efficiently solve viscous flow problems. Although the next two examples are rather idealized, they do possess the intrinsic features of more complicated boundary layer flows. The two-dimensional laminar incompressible boundary layer along a flat plate with and without pressure gradient require that one solve

$$\begin{aligned} \bar{u}\bar{u}_x + \bar{v}\bar{u}_y &= u_e u_{e_x} + \nu\bar{u}_{yy} \\ \bar{u}_x + \bar{v}_y &= 0 \end{aligned} \tag{6.12}$$

with boundary conditions

$$u(x,0) = v(x,0) = 0 \ ; \qquad u(x,\infty) = u_e \ ,$$

and initial conditions

$$u(0,y) = F(y)$$

By making the following transformation

$$\xi = x \ , \qquad \eta = y/\sqrt{2\nu x} \ , \qquad v = \bar{v}\sqrt{2\xi/\nu}$$

$$\phi = \eta u - v$$

the governing equations reduce to

(6.13a) $$2\xi u u_\xi = \phi u_\eta + u_{\eta\eta} + 2\xi u_e u_{e_\xi}$$

(6.13b) $$\phi_\eta = 2\xi u_\xi + u$$

For the case of zero pressure gradient, $u_e = 1$, the Blasius flow is obtained and the equations become independent of ξ. The solution of the ordinary differential equation that is recovered is compared with the time asymptotic solution of equations (6.13a) and (6.13b).

Due to the nonlinear term $2\xi u u_\xi$ and the decoupling of the momentum and continuity equations iteration is required for the Crank Nicolson scheme. The term $2\xi u u_\xi$ is discretized as follows

$$(2\xi U_j U_{j\xi})^{n+\frac{1}{2}} = 2\xi^{n+\frac{1}{2}} \left(\frac{U_j^{n+1} + U_j^n}{2} \right) \cdot \left(\frac{U_j^{n+1} - U_j^n}{\Delta\xi} \right) = \frac{\xi^{n+\frac{1}{2}}}{\Delta\xi} \left({U_j^{n+1}}^2 - {U_j^n}^2 \right)$$

Employing quasi-linearization (Newton-Raphson iteration)

$$\left(U_j^{n+1}\right)^2 = 2U_j^* U_j^{n+1} - \left(U_j^*\right)^2 ,$$

where U_j^* is the latest iterant, the following relationship is obtained

$$(2\xi U_j U_{j\xi})^{n+\frac{1}{2}} = \frac{\xi^{n+\frac{1}{2}}}{\Delta\xi} \left[2U_j^* U_j^{n+1} - \left(U_j^*\right)^2 - \left(U_j^n\right)^2\right]$$

Care must be taken in the choice of the initial U_j^*. Although setting $U_j^* = U_j^n$ gives a second order approximation for $\left(U_j^{n+1}\right)^2$, it reduces to first order for $(2\xi u u_\xi)^{n+\frac{1}{2}}$ due to the $\Delta\xi$ in the denominator of the derivative approximation for u_ξ.

If, however, U_j^* is approximated by the extrapolation formula

$$U_j^* = 2U_j^n - U_j^{n-1}$$

second order accuracy is recovered for the term $(2\xi U_j U_{j\xi})^{n+\frac{1}{2}}$. This procedure was employed in the Crank Nicolson calculations for both second and fourth order methods.

Since the accuracy of this linearization is of the same order as the temporal discretization error further iteration should be unnecessary. However, the decoupling of the continuity and momentum equations require iteration to obtain the desired temporal accuracy.

If $\phi u_\eta + u_{\eta\eta}$ is identified as $L(u)$ and

$$\left[(Q^n)^{-1}R^n\right]U^i \sim L^n U^i = \phi^m U^i_{\eta\eta} + U^i_\eta$$

is either the fourth order OCI approximation or the second order centered difference approximation (note that in this case $Q \equiv I$) and i and m denote appropriate time levels, then the boundary layer equations may be discretized in two ways.

Method I.

$$(6.14)\qquad 2\xi^{n+\frac{1}{2}}\, U_j^{n+\frac{1}{2}}\, (U_\xi)_j^{n+\frac{1}{2}} = \left[(Q^{n+\frac{1}{2}})^{-1}\, \frac{R^{n+\frac{1}{2}}}{\Delta\eta^2}\right]\left[\frac{U_j^{n+1} + U_j^n}{2}\right] + g(\xi^{n+\frac{1}{2}})$$

where $g(\xi)$ is the pressure gradient term. Employing Newton Raphson iteration and performing the indicated operations, the following system of equations is obtained

$$(6.15)\qquad \left[Q^{n+\frac{1}{2}}(2\xi U_j^*) - \frac{\lambda}{2} R^{n+\frac{1}{2}}\right]U_j^{n+1} = \left[Q^{n+\frac{1}{2}}\,(\xi U_j^n) - \frac{\lambda}{2} R^{n+\frac{1}{2}}\right]U_j^n$$

$$+ Q^{n+\frac{1}{2}}\left[\xi U_j^{*2} + \Delta\xi g(\xi)\right] \qquad j=1,\cdots,J-1$$

where $\lambda = \Delta\xi/\Delta\eta^2$ and ξ is evaluated at $n+\frac{1}{2}$. The system is tridiagonal and can be inverted given U at j=0 and J.

Since ϕ_j appearing in $Q^{n+\frac{1}{2}}$ and $R^{n+\frac{1}{2}}$ must be evaluated at $n+\frac{1}{2}$ the continuity equation becomes

$$(6.16)\qquad 2\xi(U_\xi)_j^{n+\frac{1}{2}} + U_j^{n+\frac{1}{2}} = (\phi_\eta)_j^{n+\frac{1}{2}}$$

or

$$(\phi_\eta)_j^{n+\frac{1}{2}} = 2\xi^{n+\frac{1}{2}}\,\frac{(U_j^{n+1} - U_j^n)}{\Delta\xi} + \frac{(U_j^{n+1} + U_j^n)}{2} = V_j^{n+\frac{1}{2}} ,$$

which can be integrated using Simpsons rule,

$$(6.17)\qquad \phi_{j+1}^{n+\frac{1}{2}} = \phi_{j-1}^{n+\frac{1}{2}} + \frac{3}{\Delta\eta}\left[V_{j+1} + 4V_j + V_{j-1}\right]^{n+\frac{1}{2}} .$$

Method II.

$$(6.18) \qquad 2\xi^{n+\frac{1}{2}}\, U_j^{n+\frac{1}{2}}\, (U_\xi)_j^{n+\frac{1}{2}} = \frac{1}{2\Delta\eta^2}\left\{\left[(Q^{n+1})^{-1} R^{n+1}\right]U_j^{n+1} + \left[(Q^n)^{-1} R^n\right]U_j^n\right\} + g(\xi^{n+\frac{1}{2}})$$

Performing the indicated operations, the following tridiagonal system of equations is obtained.

$$(6.19) \qquad \left[Q^{n+1}(2\xi^{n+\frac{1}{2}}\, U_j^*) - \frac{\lambda}{2}R^{n+1}\right]U_j^{n+1} = Q^{n+1}\left[G_j^n + \xi^{n+\frac{1}{2}}\, U_j^* + \Delta\xi g(\xi^{n+\frac{1}{2}})\right]$$

where

$$G_j^n = \frac{\lambda}{2}\left[(Q^n)^{-1}R^n\right]U_j^n + \xi^{n+\frac{1}{2}}\, U_j^n\, U_j^n \, .$$

G_j^{n+1} can be evaluated at the new time level by the recursive relations

$$(6.20) \qquad G_j^{n+1} = (\xi^{n+3/2} + 2\xi^{n+\frac{1}{2}})(U_j^{n+1})^2 - G_j^n - \Delta\xi g(\xi^{n+\frac{1}{2}}) - \xi^{n+\frac{1}{2}}(U_j^n)^2$$

The handling of the u_ξ term in the continuity equation for the above method deserves special attention. Since ϕ and u are evaluated at the (n+1) time level, u_ξ must also be evaluated there. It was thus necessary to use a one sided second order accurate derivative approximation for u_ξ (uniform mesh)

$$(6.21) \qquad (u_\xi)_j^{n+1} = (3U_j^{n+1} - 4U_j^n + U_j^{n-1})/2\Delta\xi$$

Both methods I and II worked successfully. However, method I was preferred computationally.

Lees Method

The Lees method, which does not require iteration, was also used to obtain solutions of the boundary layer equations. The discretized equation is

$$(6.22) \qquad 2\xi^n\, U_j^n\, \frac{(U_j^{n+1} - U_j^{n-1})}{2\Delta\xi} = \frac{(Q^n)^{-1}\, R^n}{3\Delta\eta^2}\left[U_j^{n-1} + U_j^n + U_j^{n+1}\right] + g(\xi^n) \, ,$$

which reduces to

$$(6.23) \qquad \left[Q^n(\xi^n\, U_j^n) - \frac{\lambda R^n}{3}\right]U_j^{n+1} = \frac{\lambda R^n}{3}\left[U_j^n + U_j^{n-1}\right] + Q^n\left[\xi^n\, U_j^n\, U_j^{n-1} + \Delta\xi g(\xi^n)\right]$$

where $\lambda = \Delta\xi/\Delta\eta^2$.

In order to update Q and R at the $(n+1) \leftarrow \xi$ level, ϕ must first be evaluated at that level. Hence it is necessary to employ (6.21) for the integration of the continuity equation.

The Lees method incorporates three levels of information, therefore a starting technique such as Crank Nicolson is needed. Furthermore, to employ variable $\Delta\xi$ requires a restarting procedure so that for such cases the Lees method may lose some of its appeal.

Table 9 compares the computed L_2 error and wall shear, τ_w, for the various methods. Note that using formula (6.6), especially for the cases with few mesh points yields better predictions for the wall shear than the standard fourth order one sided difference formula. Figure 3 which is a plot of the L_2 error versus the number of intervals illustrates the savings in storage that can be obtained by the fourth order methods.

Effects of cell Reynolds number were noted for the OCI and 2x2 block methods for the case $\Delta\eta = 1$. In the discussion below, reference will be made only to OCI since the cell Reynolds number characteristics of the block methods are not easily understood.

The local cell Reynolds number for the Blasius problem is defined as

$$R_c = \phi\Delta\eta \ .$$

The function ϕ grows linearly for large η, so that at some point, as the domain is extended, the local value of R_c will exceed $\sqrt{12}$ and when 4.207 is exceeded, as predicted by the linear analysis, oscillatory behavior will ultimately result. This conclusion indicates ironically, that in the region of small gradients (at the boundary layer edge) increased local resolution might be required to remove the oscillatory behavior. Keller [9] has made similar observations for second order methods.

In the above calculations for $\Delta\eta = 1$, with the boundary layer edge at $\eta = 6$ no oscillations occurred (with the internal $(R_c)_{max} = 3.790$ at $\eta = 5.0$). However, with the boundary layer edge extended to $\eta = 10.0$, oscillations appeared from the outer boundary inward to $\eta = 8$. Since the velocity oscillates (< 1.5%) about a value greater than 1, in the region $4 \leq \eta < 10$, the velocity profile exhibits overshoots.

Enlarging the domain does not cause the oscillations to invade the region where the cell Reynolds number condition is satisfied as inspection of the solution reveals. Furthermore, the computed wall shear is affected only slightly (see below).

η_∞	τ_w (Eq. 6.6)
6.0	.4771
10.0	.4869
20.0	.4893

The second boundary layer example considered was the flow with an adverse pressure gradient, the Howarth problem, where the external velocity is linearly retarded

$$u_e = 1 - \xi \ .$$

As a consequence of the adverse pressure gradient, separation (vanishing of the wall shear) will occur at some point downstream along the flat plate. The determination of the entire flow development, from the initial Blasius profile to the point of separation is sought. Since the boundary layer equations break down at separation, it is expected that the numerical computations will be sensitive near that point. Factors such as $\Delta\xi$ step size and iteration criteria may strongly influence the calculations. A complete analysis of the behavior of the numerical solution near the separation point is not considered in this report. However, a set of calculations for a fixed $\Delta\xi = 10^{-4}$ and iteration convergence parameter, $\varepsilon = 10^{-5}$ have been obtained and are shown in Table 10. The computed separation point (the point where the shear changed sign) along with the running time for each calculation is given.

In general, one iteration was required for convergence of the Crank Nicolson discretization, except near separation. Several calculations did not converge near the separation point and were thus terminated there. These are indicated by asterisks in Table 10.

Hirsh [6] employing the 3x3 block method computed the separation point at $\xi = .119818$ for $\Delta\eta = .2$. Since no discussion of the behavior of the velocity profiles near separation was given in [6] a detailed comparison can not be made. However, our calculations show that the second order Crank Nicolson and the 2x2 block methods give $\xi_{sep} > .1200$ for all $\Delta\eta$, whereas OCI (in particular the Lees discretization) yields values of $\xi_{sep} < .1200$.

Near separation the cell Reynolds number becomes very large and exceeds the limits set by the linear theory. However, the actual behavior of the solution in this region is not predicted well, and simple monitoring of R_c is not helpful as it was for the Blasius problem since it is now necessary to take into account the nonhomogeneous terms contribution. Further investigation is necessary to understand even the nonhomogeneous linear case.

Table 6.1

Matrix Setup and Inversion Operations* Uniform Mesh

MATRIX	INVERSION		SETUP	TOTAL
	ESTIMATED**	ACTUAL		
Scalar Tridiagonal (OCI-CN)	5N-4	5N-4	22N-22	27N-26
2x2 Block Tridiagonal (C-N)	36N-24	27N-60	8N+16	35N-44
3x3 Block Tridiagonal (C-N)	108N-72	49N-62	4N+24	53N-38
Scalar Pentadiagonal	11N-16		10N ***	21N-16

* Here it is assumed that multiplications and divisions are equivalent. However on certain machines this may not be true, e.g. on the CDC 6600 a division is comparable to six multiplications. The operation counts would have to be changed accordingly for the methods.

** *Reference* [8].

*** Does not include extrapolation formulas for points adjacent to the boundaries.

Table 6.2

Linear Variable Coefficient Parabolic Equation

$$u_t = a(x,t)u_{xx} + b(x,t)u_x$$
$$u = \exp\{(x+1)(t+2)\}$$

METHOD	N	TIME STEPS	L_2 ERROR	L_2 RATE	COMPUTING TIME* (SEC)
Second Order Crank Nicolson	100	2000	$.20 * 10^{-04}$		35.5
				1.98	
	160	2000	$.79 * 10^{-05}$		55.4
				1.96	
	200	2000	$.51 * 10^{-05}$		68.8
				1.97	
	400	2000	$.13 * 10^{-05}$		135.5
3x3 Block Crank Nicolson	5	2000	$.15 * 10^{-04}$		12.2
				4.06	
	10	2000	$.90 * 10^{-06}$		17.7
				4.14	
	20	2000	$.51 * 10^{-07}$		30.4
				4.67	
	40	2000	$.20 * 10^{-08}$		58.5
2x2 Block Crank Nicolson	5	2000	$.83 * 10^{-05}$		7.4
				3.58	
	10	2000	$.70 * 10^{-06}$		11.9
				3.87	
	20	2000	$.48 * 10^{-07}$		20.8
				4.59	
	40	2000	$.20 * 10^{-08}$		40.2
Operator Compact Implicit Crank Nicolson	5	2000	$.24 * 10^{-04}$		5.4
				4.00	
	10	2000	$.15 * 10^{-05}$		8.8
				4.00	
	20	2000	$.94 * 10^{-07}$		15.6
				4.52	
	40	2000	$.41 * 10^{-08}$		30.0

* Computation times are for a CDC 6500

Table 6.3

Linear Variable Coefficient Parabolic Equation

$$u_t = a(x,t)u_{xx} + b(x,t)u_x$$

$$u = u_e = \exp(x+1)(t+2)$$

$u(0) + u_x(0) = (t+1) \exp [t+2]$, $u(1) = u_e(1)$

OCI - 2000 TIME STEPS

N	L_2 ERROR	L_2 RATE	COMPUTING TIME* (SEC)
5	$.222 * 10^{-02}$		6.57
		4.796	
10	$.122 * 10^{-03}$		9.58
		4.049	
20	$.737 * 10^{-05}$		16.71
		4.063	
40	$.441 * 10^{-06}$		30.42

* Computation times for a CDC 6500

Table 6.4

Steady State Solution of Burgers Equation
Second Order Crank Nicolson

ν	N	DX	$\nu DT/DX^2$	MAX ERROR	L_2 ERROR	L_2 RATE
.500	50	.20	6.25	$.633 * 10^{-3}$	$.125 * 10^{-2}$	2.007
	100	.10	25.00	$.158 * 10^{-3}$	$.311 * 10^{-3}$	1.999
	200	.05	100.00	$.395 * 10^{-4}$	$.778 * 10^{-4}$	
.250	50	.20	3.125	$.303 * 10^{-2}$	$.442 * 10^{-2}$	2.020
	100	.10	12.50	$.747 * 10^{-3}$	$.109 * 10^{-2}$	1.997
	200	.05	50.00	$.186 * 10^{-3}$	$.273 * 10^{-3}$	
.125	50	.20	1.5625	$.128 * 10^{-1}$	$.131 * 10^{-1}$	2.061
	100	.10	6.25	$.303 * 10^{-2}$	$.314 * 10^{-2}$	2.019
	200	.05	25.00	$.749 * 10^{-3}$	$.775 * 10^{-3}$	
.062	50	.20	.775	$.694 * 10^{-1}$	$.473 * 10^{-1}$	2.331
	100	.10	3.100	$.130 * 10^{-1}$	$.940 * 10^{-2}$	2.069
	200	.05	12.40	$.308 * 10^{-2}$	$.224 * 10^{-2}$	
.031	100	.10	1.55	$.694 * 10^{-1}$	$.334 * 10^{-1}$	2.328
	200	.05	6.20	$.130 * 10^{-2}$	$.665 * 10^{-2}$	

Table 6.5

Steady State Solution of Burgers Equation
OCI Crank Nicolson & Lees

ν	N	DX	$\nu DT/DX^2$	MAX ERROR	L_2 ERROR	L_2 RATE
.500	10	1.00	.25	$.132 * 10^{-2}$	$.231 * 10^{-2}$	
	20	.50	2.00	$.796 * 10^{-4}$	$.137 * 10^{-3}$	4.076
	50	.20	6.25	$.205 * 10^{-5}$	$.348 * 10^{-5}$	4.028
	100	.10	25.00	$.128 * 10^{-6}$	$.216 * 10^{-6}$	3.985
.250	10	1.00	.125	$.189 * 10^{-1}$	$.267 * 10^{-1}$	
	20	.50	.500	$.126 * 10^{-2}$	$.153 * 10^{-2}$	4.125
	50	.20	3.125	$.312 * 10^{-4}$	$.370 * 10^{-4}$	4.062
	100	.10	12.500	$.194 * 10^{-5}$	$.230 * 10^{-5}$	4.008
.125	20	.5	.250	$.187 * 10^{-1}$	$.188 * 10^{-1}$	
	50	.20	1.563	$.466 * 10^{-3}$	$.431 * 10^{-3}$	4.120
	100	.10	6.250	$.312 * 10^{-4}$	$.261 * 10^{-4}$	4.046
.062	50	.20	.388	$.868 * 10^{-2}$	$.554 * 10^{-2}$	
	100	.10	3.100	$.484 * 10^{-3}$	$.313 * 10^{-3}$	4.145
.031	60	.167	.558	$.598 * 10^{-1}$	$.346 * 10^{-1}$	
	100	.10	1.550	$.868 * 10^{-2}$	$.392 * 10^{-2}$	4.263

Table 6.6

Steady State Solution of Burgers Equation
Comparison of U Profiles

$\nu = .500$

X	EXACT U	OCI - C-N			2ND ORDER C-N
		N = 10	N = 20	N = 100	N = 200
-5.00	.993307	.993307	.993307	.993307	.993307
-4.00	.982014	.982042	.982015	.982014	.982021
-3.00	.952574	.952845	.952589	.952574	.952595
-2.00	.880797	.881716	.880850	.880797	.880833
-1.00	.731059	.732380	.731138	.731059	.731094
- .40	.598688			.598688	.598705
- .20	.549834			.549834	.549842
-0.00	.500000	.500000	.500000	.500000	.500000

Table 6.7

Steady State Solution of Burgers Equation
Comparison of U Profiles

$\nu = .031$

X	EXACT U	OCI - CN N = 60	OCI - CN N = 100	2ND ORDER CN N = 200
-1.200	1.000000		1.000000	1.000000
-1.167	1.000000	.999997		
-1.000	1.000000	.999990	1.000000	1.000000
- .833	.999999	.999963		
- .800	.999998		.999995	1.000000
- .667	.999979	.999894		
- .600	.999937		.999903	1.000000
- .500	.999086	.999651		
- .400	.998425		.998091	.999981
- .333	.995397	.998843		
- .200	.961794		.962779	.994937
- .167	.936325	.996115		
0.000	.500000	.500000	.500000	.500000
.167	.063675	.003885		
.200	.038206		.037221	.005062
.333	.004603	.001157		
.400	.001575		.001909	.000019
.500	.000314	.000349		
.600	.000063		.000097	.000000
.667	.000021	.000106		
.800	.000002		.000005	.000000
.833	.000001	.000032		
1.000	.000000	.000010	.000000	.000000
1.167	.000000	.000003		
1.200	.000000		.000000	.000000

Table 6.8

Two Dimensional Parabolic Equation

OCI - Crank-Nicolson

$$u_t = a(x,y,t)\, u_{xx} + b(x,y,t)\, u_x + c(x,y,t)\, u_{yy} + d(x,y,t)\, u_y$$

$$u(x,y,t) = \exp\{(x+1)(y+1)(t+1)\}$$

Domain is square $\Omega = [1/2 \le x,y \le 1]$, $\Delta x = \Delta y = h$

TIME STEPS	h	Δt	L_2-ERROR	L_2-RATE	MAX RELATIVE ERROR	MAX RELATIVE RATE
5	.1	.1	3.235-03		1.544-04	
				4.414		3.905
20	.05	.025	1.517-04		1.031-05	
				4.955		3.977
80	.025	.00625	4.890-06		6.549-07	
10	.1	.1	3.903-02		3.909-04	
				4.364		3.933
40	.05	.025	1.896-03		2.559-05	
				4.909		3.982
160	.025	.00625	6.311-05		1.619-06	

Table 6.9

Blasius Flow

METHOD	N	$\Delta\eta$	L_2 ERROR	L_2 RATE	τ_w^* WALL SHEAR	
Second Order C-N	50	.20	$.173 * 10^{-2}$		.470325 **	
				2.00		
	100	.10	$.437 * 10^{-3}$		.469726	
				2.00		
	200	.05	$.108 * 10^{-3}$		.469625	
Fourth Order 2x2 Block	10	1.00	$.450 * 10^{-2}$		.428199***	
				5.49		
	20	.50	$.100 * 10^{-3}$		.467035	
				5.27		
	50	.20	$.259 * 10^{-5}$		.469586	
				3.98		
	100	.10	$.164 * 10^{-6}$		.4695997	
					+	++
Fourth Order OCI C-N & Lees	10	1.00	$.279 * 10^{-1}$		.437665	.485913
				7.34		
	20	.50	$.172 * 10^{-3}$		.465537	.469696
				5.24		
	50	.20	$.456 * 10^{-5}$		.469577	.4695996
				4.00		
	100	.10	$.285 * 10^{-6}$		.4695996	.4695999

* Exact value [21] $\tau_w = .469600$

** Computed by second order three point end difference formula

*** Hamming formula used at boundary

\+ Computed by fourth order six point end difference formula

++ Computed by Eq. (6.6)

Table 6.10

Howarth Flow

METHOD	N	$\Delta\eta$	ξ_{sep}^{+}	RUNNING TIME (SEC)
Second Order Crank Nicolson	20	.50	.1252	8.90
	50	.20	.1209	19.92
	100	.10	.1204	38.41
	200	.05	.1202	71.58
2x2 Block Tridiagonal Crank Nicolson	20	.50	.1202*	19.38
	50	.20	.1202	47.47
	100	.10	.1202	94.05
OCI Crank Nicolson	20	.50	.1188*	13.74
	50	.20	.1199*	35.81
	100	.10	.1201*	73.63
OCI Lees	20	.50	.1186**	11.66
	50	.20	.1200	27.55
	100	.10	.1200	54.21

$\Delta\xi = 10^{-4}$, $\varepsilon^{***} = 10^{-5}$

+ Point where the shear changed sign

* Calculation didn't converge and was terminated

** Oscillations occurred and calculation was terminated

*** Iteration parameter

REFERENCES

1. Y. Adam, "A Hermitian Finite Difference Method for the Solution of Parabolic Equations," Comp. & Maths. with Appls., Vol. 1 pp. 393-406, Pergamon Press (1975).

2. W. R. Briley and H. McDonald, "Solution of the Multidimensional Compressible Navier-Stokes Equations by a Generalized Implicit Method," United Technologies R75-911363-15, (January 1976).

3. M. Ciment, S. H. Leventhal, "Higher Order Compact Implicit Schemes for the Wave Equation," Math. of Comp. Vol. 29, No. 132, 1975, pp. 985-994.

4. M. Ciment and S. H. Leventhal, "A Note on the Operator Compact Implicit Method for the Wave Equation," NSWC/WOL/TR 76-153 (November 1976).

5. L. Collatz, The Numerical Treatment of Differential Equations, Springer-Verlag, Berlin, (1960).

6. R. S. Hirsh, "Higher Order Accurate Difference Solution of Fluid Mechanics Problems by a Compact Differencing Technique," J. Computational Physics, Vol. 19, pp. 90-109, (1975).

7. R. S. Hirsh and D. H. Rudy, "The Role of Diagonal Dominance and Cell Reynolds Number in Implicit Difference Methods for Fluid Mechanics Problems," J. Computational Physics, Vol. 25, pp. 304-310, (1974).

8. E. Isaacson and H. B. Keller, Analysis of Numerical Methods, Wiley, New York, (1966).

9. Keller, H. B., Numerical Methods for Two-Point Boundary Value Problems, Blaisdell Publishing Co., Waltham, Mass., (1968).

10. Krause, E., Hirschel, E. H. and Kordulla, W., "Fourth Order "Mehrstellen" - Integration for Three Dimensional Turbulent Boundary Layers," Computers and Fluids, Vol. 4, pp. 77-92, (1976).

11. D. C. L. Lam and R. B. Simpson, "Centered Differencing and the Box Scheme for Diffusion Convection Problems," J. Comp. Physics, Vol. 22, pp. 486-500, (1976).

12. M. Lees, "A Linear Three-Level Difference Scheme for Quasilinear Parabolic Equations," Math Comp., Vol. 20, pp. 516-522, (1966).

13. Mitchell, A. R., Computational Methods in Partial Differential Equations, John Wiley, London, (1969).

14. S. A. Orszag and M. Israeli, "Numerical Simulation of Viscous Incompressible Flows," Annual Review of Fluid Mechanics," Vol. 6, (1974).

15. J. M. Ortega, Numerical Analysis, Academic Press, New York (1972).

16. J. M. Ortega and W. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, (1970).

17. Peters, N., "Boundary Layer Calculations by a Hermitian Finite Difference Method," Fourth International Conference on Numerical Methods in Fluid Mechanics, Boulder, Colorado, Springer-Verlag, Berlin (1974).

18. A. Ralston, A First Course in Numerical Analysis, McGraw-Hill, New York, (1965).

19. R. D. Richtmyer and K. W. Morton, Difference Methods for Initial-Value Problems 2nd Ed., Interscience Publishers, New York, (1967).

20. P. J. Roache, Computational Fluid Dynamics, Hermosa Publishers, Albuquerque, (1972).

21. Rosenhead, L., Laminar Boundary Layers, Oxford University Press, London, (1963).

22. S. G. Rubin and P. K. Khosla, "Higher Order Numerical Methods Derived from Three-Point Polynomial Interpolation," NASA CR-2735, (August 1976).

23. Rubin, S. G. and Graves, R. A., Jr., "A Cubic Spline Approximation for Problems in Fluid Dynamics," Old Dominion University, TR 74-T1, Norfolk, Va., (June 1974).

24. B. K. Swartz, "The Construction of Finite Difference Analogs of Some Finite Element Schemes," Mathematical Aspects of Finite Elements in Partial Differential Equations (C. DeBoor, Ed.) Academic Press, pp. 279-312, (1974).

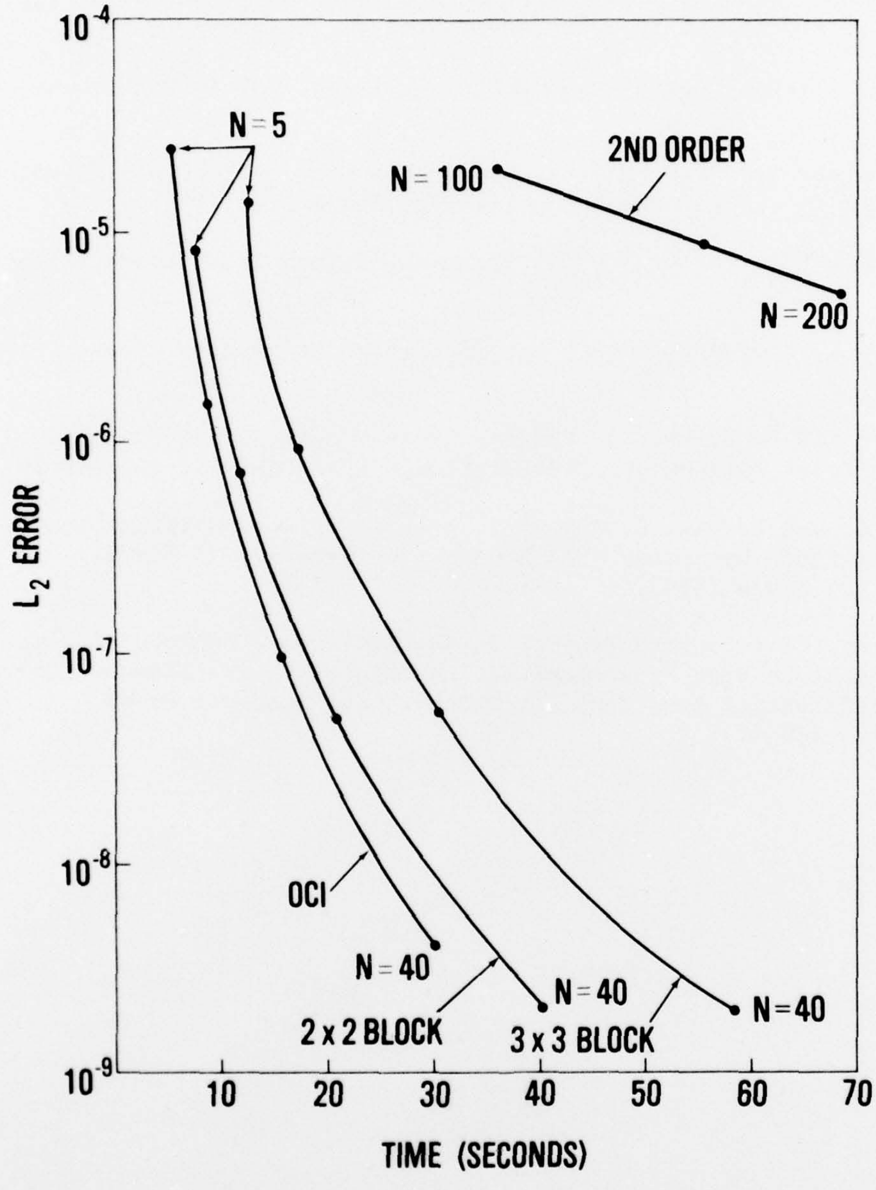


FIG. 1 LINEAR VARIABLE COEFFICIENT EQUATION - L_2 ERROR VS RUNNING TIME

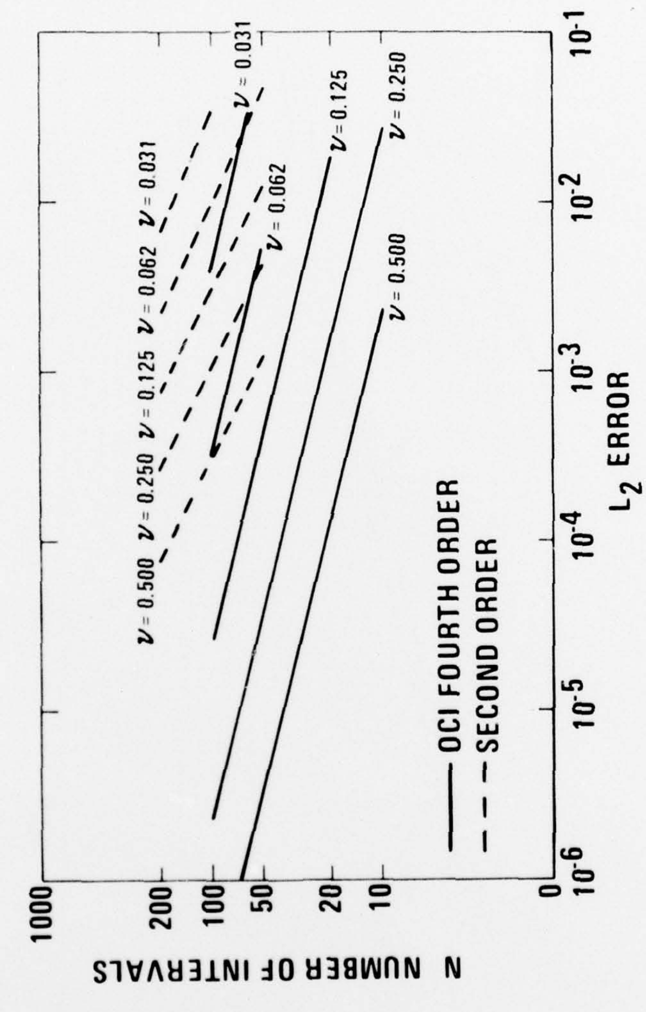


FIG. 2 STEADY STATE BURGERS EQUATION - L_2 ERROR VS NUMBER OF INTERVALS

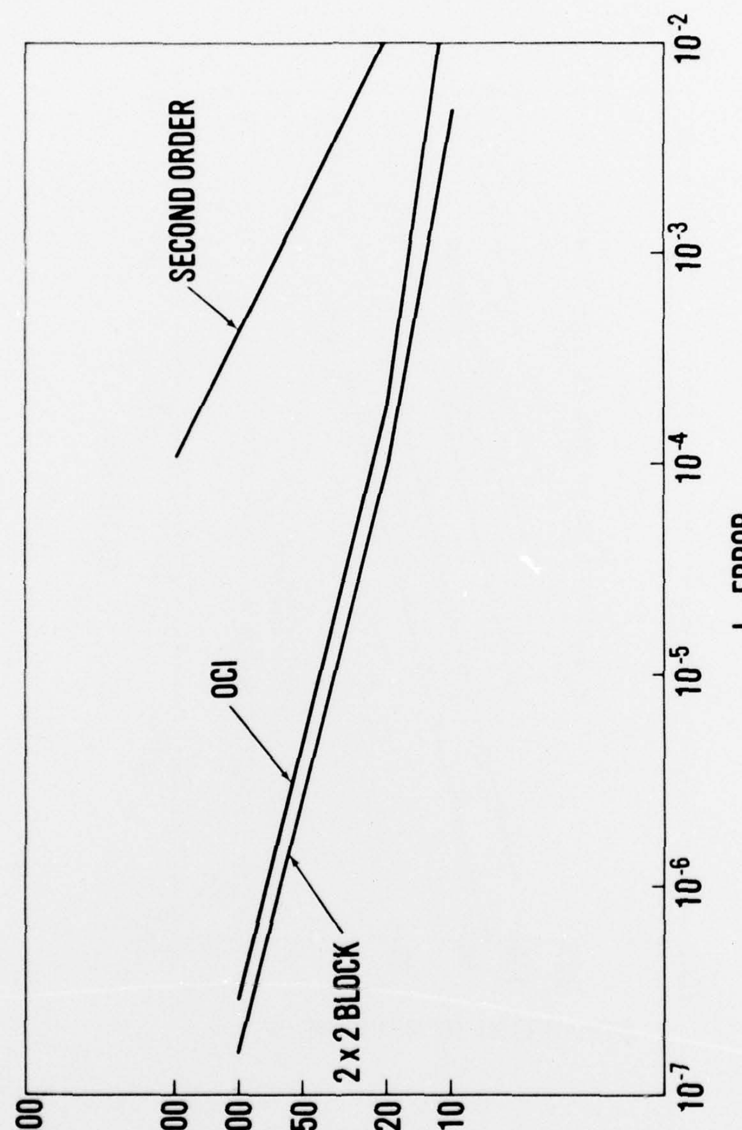


FIG. 3 BLASIUS FLOW - L_2 ERROR VS NUMBER OF INTERVALS

APPENDIX A

The operator compact implicit formulas are derived here for uniform and nonuniform grids, with their associated truncation errors.

Given

$$\text{(A-1)} \qquad Lu = au_{xx} + bu_x ,$$

a linear relationship between u and the spatial operator, Lu, at x_j is sought in the form

$$\text{(A-2)} \qquad r^- u_- + r^0 u_0 + r^+ u_+ = q^- L(u)_- + q^0 L(u)_0 + q^+ L(u)_+$$

where as shorthand notation the subscripts -, 0, + are used for j-1, j and j+1 respectively, and the j dependence of the coefficients is not indicated, see (2.12).

The function values u_- and u_+ and the spatial operators $L(u)_-$ and $L(u)_+$ can be obtained through Taylor's series expansion about the point j.

$$\text{(A-3a)} \qquad u_+ = u_0 + h_+ u_0^{(1)} + \frac{h_+^2}{2!} u_0^{(2)} + \frac{h_+^3}{3!} u_0^{(3)} + \frac{h_+^4}{4!} u_0^{(4)} + \frac{h_+^5}{5!} u_0^{(5)} + \frac{h_+^6}{6!} u_0^{(6)} + \cdots$$

$$\text{(A-3b)} \qquad u_- = u_0 - h_- u_0^{(1)} + \frac{h_-^2}{2!} u_0^{(2)} - \frac{h_-^3}{3!} u_0^{(3)} + \frac{h_-^4}{4!} u_0^{(4)} - \frac{h_-^5}{5!} u_0^{(5)} + \frac{h_-^6}{6!} u_0^{(6)} - \cdots$$

$$\text{(A-3c)} \qquad L(u)_- = a_- u_-^{(2)} + b_- u_-^{(1)} = b_- u_0^{(1)} + (a_- - h_- b_-)u_0^{(2)}$$
$$- h_-\left(a_- - \frac{h_-}{2} b_-\right)u_0^{(3)} + \frac{h_-^2}{2!}\left(a_- - \frac{h_-}{3} b_-\right)u_0^{(4)}$$
$$- \frac{h_-^3}{3!}\left(a_- - \frac{h_-}{4} b_-\right)u_0^{(5)} + \frac{h^4}{4!}\left(a_- - \frac{h_-}{5} b_-\right)u_0^{(6)} - \cdots$$

$$\text{(A-3d)} \qquad L(u)_+ = a_+ u_+^{(2)} + b_+ u_+^{(1)} = b_+ u_0^{(1)} + (a_+ + h_+ b_+)u_0^{(2)}$$
$$+ h_+\left(a_+ + \frac{h_+}{2} b_+\right) u_0^{(3)} + \frac{h_+^2}{2!}\left(a_+ + \frac{h_+}{3} b_+\right)u_0^{(4)}$$
$$+ \frac{h_+^3}{3!}\left(a_+ + \frac{h_+}{4} b_+\right) u_0^{(5)} + \frac{h_+^4}{4!}\left(a_+ + \frac{h_+}{5} b_+\right) u_0^{(6)} + \cdots$$

where superscripts in parenthesis indicate derivatives and $h_+ = x_{j+1} - x_j$ and $h_- = x_j - x_{j-1}$. Multiplying equations (A-3a) - (A-3d) by α, β, γ, δ, respectively and collecting terms the following relation is obtained

(A-4) $$\alpha u_+ + \beta u_- + \gamma L(u)_- + \delta L(u)_+ = (\alpha+\beta)u_0 + \overline{B}\, u_0^{(1)} + \overline{A}\, u_0^{(2)} + \overline{C}\, u_0^{(3)} + \overline{D}\, u_0^{(4)}$$

or

(A-5) $$\alpha u_+ - (\alpha+\beta)u_0 + \beta u_- = -\gamma L(u)_- - \delta L(u)_+ + \overline{A}\, u_0^{(2)} + \overline{B}\, u_0^{(1)} + \overline{C}\, u_0^{(3)} + \overline{D}\, u_0^{(4)}$$

$$+ \text{Truncation Error},$$

where in order that (A-5) be equal to (A-2),

(A-6)
$$\overline{B} \equiv \alpha h_+ - \beta h_- + \delta b_+ + \gamma b_- = b_0$$

$$\overline{A} \equiv \frac{\alpha h_+^{\ 2}}{2} + \frac{\beta h_-^{\ 2}}{2} + \delta\left(a_+ + h_+ b_+\right) + \gamma\left(a_- - h_- b_-\right) = a_0$$

$$\overline{C} \equiv \frac{\alpha h_+^{\ 3}}{3!} - \frac{\beta h_+^{\ 3}}{3!} + \delta h_+\left(a_+ + \frac{h_+}{2} b_+\right) - \gamma h_-\left(a_- - \frac{h_-}{2} b_-\right) = 0$$

$$\overline{D} \equiv \frac{\alpha h_+^{\ 4}}{4!} + \frac{\beta h_-^{\ 4}}{4!} + \frac{\delta h_+^{\ 2}}{2}\left(a_+ + \frac{h_+}{3} b_+\right) + \frac{\gamma h_-^{\ 2}}{2}\left(a_- - \frac{h_-}{3} b_-\right) = 0$$

Define

(A-7) $$\hat{\alpha} = \alpha D,\ \hat{\beta} = \beta D,\ \hat{\gamma} = \gamma D,\ \hat{\delta} = \delta D$$

where D is the determinant of (A-6)

(A-8)
$$\begin{aligned} D = \Big\{ & 12a_-a_+(h_+^{\ 3} + 4h_+^{\ 2}\, h_- + 4h_+\, h_-^{\ 2} + h_-^{\ 3}) \\ & - 2a_+b_-h_-(3h_+^{\ 3} + 7h_+^{\ 2}\, h_- + 5h_+\, h_-^{\ 2} + h_-^{\ 3}) \\ & + 2a_-b_+h_+(h_+^{\ 3} + 5h_+^{\ 2}\, h_- + 7h_+\, h_-^{\ 2} + 3h_-^{\ 3}) \\ & - h_+h_-b_+b_-(h_+ + h_-)^3 \Big\}, \end{aligned}$$

then the variables $\hat{\alpha}$, $\hat{\beta}$, $\hat{\gamma}$ and $\hat{\delta}$ are given by

(A-9)
$$\begin{aligned} \hat{\gamma} = \Big\{ & 12h_+\, a_+\, a_0\, (h_+^{\ 2} - h_+\, h_- - h_-^{\ 2}) \\ & + 2a_+b_0h_+^{\ 2}h_-(3h_+ + 2h_-) + 2a_0b_+h_+^{\ 2}(h_+^{\ 2} - h_+h_- - 2h_-^{\ 2}) \\ & + h_+^{\ 3}h_-b_0b_+(h_+ + h_-) \Big\} \end{aligned}$$

(A-10) $$\hat{\delta} = \left\{12a_0a_-h_-(h_-^2 - h_+h_- - h_+^2) + 2a_0b_-h_-^2(2h_+^2 + h_+h_- - h_-^2)\right.$$

$$\left. - 2a_-b_0h_+h_-^2(2h_+ + 3h_-) + b_0b_-h_+h_-^3(h_+ + h_-)\right\}$$

(A-11) $$\hat{\beta}h_-(h_+ + h_-) = D\left[2a_0 - h_+b_0\right] - \hat{\delta}\left[2a_+ + h_+b_+\right] - \hat{\gamma}\left[2a_- - b_-(2h_- + h_+)\right]$$

(A-12) $$\hat{\alpha}h_+(h_+ + h_-) = D\left[2a_0 + h_-b_0\right] - \hat{\delta}\left[2a_+ + b_+(2h_+ + h_-)\right] - \hat{\gamma}\left[2a_- - h_-b_-\right]$$

Multiplying thru by -D, the q's and r's become

(A-13) $$q^- = \hat{\gamma},\ q^+ = \hat{\delta},\ q^0 = -D$$

$$r^- = -\hat{\beta},\ r^0 = (\hat{\alpha} + \hat{\beta}),\ r^+ = -\hat{\alpha}$$

such that the operators Q and R are given in the form

(A-14) $$Q = \hat{\delta}\ S_+ - D\ I + \hat{\gamma}\ S_-$$

$$R = -\ \hat{\alpha}S_+ + (\hat{\alpha} + \hat{\beta})I - \hat{\beta}\ S_-$$

where S is the shift operator.

Using the relations (A-7) - (A-12) the truncation error given by

(A-15) $$E_T = \frac{1}{D}\left\{\frac{\hat{\alpha}h_+^5}{5!} - \frac{\hat{\beta}h_-^5}{5!} + \frac{\hat{\delta}h_+^3}{3!}\left(a_+ + \frac{h_+}{4}b_+\right) - \frac{\hat{\gamma}h_-^3}{3!}\left(a_- - \frac{h_-}{4}b_-\right)\right\}u_0^{(5)} +$$

$$\frac{1}{D}\left\{\frac{\hat{\alpha}h_+^6}{6!} + \frac{\hat{\beta}h_-^6}{6!} + \frac{\hat{\delta}h_+^4}{4!}\left(a_+ + \frac{h_+}{5}b_+\right) + \frac{\hat{\gamma}h_-^4}{4!}\left(a_- - \frac{h_-}{5}b_-\right)\right\}u_0^{(6)}$$

is seen to be third order accurate for small h.

For a uniform mesh, $h_+ = h_- = h$, the truncation error reduces to

(A-16) $$E_T = \frac{h^4}{1800\ a_+a_-}\left\{-\left[9\ a_0a_-a_+\right]u_0^{(6)} + \left[4\ a_0a_-b_+\right.\right.$$

$$\left.\left. - 35\ a_-a_+b_0 + 4\ a_0a_+b_-\right]u_0^{(5)}\right\}$$

which is fourth order accurate.

Note that in equation (2.12), common factors in the q's and r's have been canceled (involving constant h), so that (2-12) differs from (A-13) by a multiplicative constant, $2h^3$.

APPENDIX B

The coefficients of (6.6) namely

(B-1) $$F_0 \equiv (u_x)_0 = H_0U_0 + H_1U_1 + H_2U_2 + G_0f_0 + G_1f_1 + G_2f_2$$

are derived.

Consider the compact implicit formulas

(B-2a) $$F_0 + 4F_1 + F_2 = \frac{3}{h}(U_2 - U_0)$$

(B-2b) $$S_0 + 10S_1 + S_2 = \frac{12}{h^2}(U_0 - 2U_1 + U_2)$$

(B-2c) $$S_0 + 4S_1 + S_2 = \frac{3}{h}(F_2 - F_0)$$

and the differential equation at points j=0,1,2 expressed as

(B-3) $$\alpha_j S_j + b_j F_j = f_j \qquad j=0,1,2 \; .$$

Equations (B-2) - (B-3) form a system of 6 equations in 9 unknowns, and thus F_0 can be determined as a function of u_0, u_1, u_2, f_0, f_1 and f_2.

The coefficients in (B-1) are listed below.

$$H_0 = \left\{ - \left(\frac{6}{h}\right)\left(\frac{b_1}{a_1}\right)\left(3\frac{b_2}{a_2} + \frac{15}{h}\right) + \frac{48}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)\right\}/C$$

$$H_1 = - \frac{96}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)/C$$

$$H_2 = \left\{\left(\frac{6}{h}\right)\left(\frac{b_1}{a_1}\right)\left(3\frac{b_2}{a_2} + \frac{15}{h}\right) + \frac{48}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)\right\}/C$$

(B-4) $$G_0 = \frac{-2}{a_0}\left(\frac{3b_1}{a_1} + \frac{6}{h}\right)/C$$

$$G_1 = \frac{-8}{a_1}\left(3\frac{b_2}{a_2} + \frac{15}{h}\right)/C$$

$$G_2 = \frac{-2}{a_2}\left(\frac{3b_1}{a_1} + \frac{6}{h}\right)/C$$

$$C = 2\left\{3\frac{b_1}{a_1}\left(\frac{b_2}{a_2} - \frac{b_0}{a_0}\right) + \frac{6}{h}\left(5\frac{b_1}{a_1} - \frac{b_2}{a_2} - \frac{b_0}{a_0}\right)\right\}$$

The truncation error is given by

$$E_{TRUNC} = \left\{ -\left(4\frac{b_3}{a_3} - 10\frac{b_2}{a_2}\right)E_S + \left[4\frac{b_3}{a_3}\left(\frac{b_3}{a_3} - \frac{b_2}{a_2} + \frac{3}{h}\right) - \left(\frac{b_3}{a_3} + \frac{3}{h}\right)\left(4\frac{b_3}{a_3} - 10\frac{b_2}{a_2}\right)\right]E_F + 4\left(\frac{b_3}{a_3} - \frac{b_2}{a_2} + \frac{3}{h}\right)E_T \right\}$$

where

$$E_S = 5\frac{h^4}{30}u^{(6)}, \quad E_F = \frac{h^4}{30}u^{(5)}, \quad E_T = \frac{h^4}{20}u^{(6)}$$

Eq. (B-5) can be specialized for constant coefficients

(B-6) $$E_{TRUNC} = \frac{h^5}{\left(\frac{b}{a}\right) 180}\left[\left(5\frac{b}{a} + \frac{3}{h}\right)u^{(6)} + \left(\frac{b}{a} + \frac{5}{h}\right)u^{(5)}\right]$$

In the case of time dependent problems modifications to (B-1) are necessary. Consider the one dimensional parabolic equation

(B-7) $$u_t = L(u) \equiv aS + bF = f$$

The first derivative at an end point at time level (n+1) in the form of

(B-8) $$F_0^{n+1} = H_0^{n+1} U_0^{n+1} + H_1^{n+1} U_1^{n+1} + H_2^{n+1} U_2^{n+1} + G_0^{n+1} f_0^{n+1} + G_1^{n+1} f_1^{n+1} + G_2^{n+1} f_2^{n+1}$$

is sought.

Again, as before use the compact implicit formulas (B-2), but with u,F and S evaluated at time level (n+1). The differential equation (B-7), however, discretized temporally by a Crank Nicolson scheme to yield

(B-9) $$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \left[a_j^{n+1} S_j^{n+1} + b_j^{n+1} F_j^{n+1}\right]/2 + \left[a_j^n S_j^n + b_j^n F_j^n\right]/2 \equiv \frac{f_j^{n+1} + f_j^n}{2}$$

Thus f appearing in (B-8) is the spatial operator evaluated at (n+1) and is given by

$$f_j^{n+1} \equiv a_j^{n+1} S_j^{n+1} + b_j^{n+1} = \frac{2}{\Delta t} U_j^{n+1} - \left(f_j^n + \frac{2}{\Delta t} U_j^n\right) \tag{B-10}$$

Hence, substituting (B-10) into (B-8), the desired relationship is obtained

$$\begin{aligned} F_0^{n+1} = {} & \left[H_0^{n+1} + \frac{2}{\Delta t} G_0^{n+1}\right] U_0^{n+1} + \left[H_1^{n+1} + \frac{2}{\Delta t} G_1^{n+1}\right] U_1^{n+1} \\ & + \left[H_2^{n+1} + \frac{2}{\Delta t} G_1^{n+1}\right] U_2^{n+1} + G_0^{n+1}\left[f_0^n + \frac{2}{\Delta t} U_0^n\right] \\ & + G_1^{n+1}\left[f_1^n + \frac{2}{\Delta t} U_1^n\right] + G_2^{n+1}\left[f_2^n + \frac{2}{\Delta t} U_2^n\right] \end{aligned} \tag{B-11}$$

The local spatial truncation error remains unchanged.